CK.0000062869

NGUYỄN VIẾT ĐẢM

Mô phỏng hệ thống viễn thông và ứng dụng Matlab

(Có kèm đĩa CD-ROM)

UYÊN
LIỆU

NHÀ XUẤT BẢN BƯU ĐIỆN

NGUYỄN VIẾT ĐẢM

Mô phỏng hệ thống viễn thông và ứng dụng Matlab

ĐẠI HỌC THÁI NGUYÊN
TRUNG TÂM HỌC LIỆU

NHÀ XUẤT BẢN BƯU ĐIỆN
Hà Nội - 2007

Mã số: HT 02 HM 07

LỜI NÓI ĐẦU

Mô phỏng là một quá trình bắt chước hay phỏng tạo, dùng để nghiên cứu các mối quan hệ giữa các tham số tương tác trong hệ thống. Từ quan điểm thiết kế và phân tích hiệu năng hệ thống, bằng cách dung hoà giữa tính phức tạp của hệ thống với khả năng xử lý của thiết bị, bài toán thiết kế dựa vào mô phỏng được xây dựng và áp dụng ngày càng rộng rãi trong lĩnh vực viễn thông. Thông qua mô phỏng, cho phép nghiên cứu các đặc trưng hoạt động của hệ thống phức tạp hơn, thực tế hơn so với những hệ thống không dựa trên mô phỏng, bởi lẽ các hệ thống không dựa vào mô phỏng tính phức tạp bị hạn chế để đảm bảo việc phân tích có thể kiểm soát được. Mô phỏng cho phép thay đổi một cách dễ dàng các thông số hệ thống và nhanh chóng đánh giá được các ảnh hưởng của việc thay đổi đó ở dạng hiển thị trực quan và tương tác của các kết quả mô phỏng.

Với hy vọng cung cấp cho bạn đọc những kiến thức cần thiết về phương pháp luận mô phỏng, lập mô hình mô phỏng và triển khai mô hình môi phỏng trong quá trình tiết kế tối ưu hệ thống viễn thông, Nhà xuất bản Bưu điện xuất bản cuốn sách ***"Mô phỏng hệ thống viễn thông và ứng dụng Matlab"*** của tác giả Nguyễn Viết Đảm, giảng viên Học viện Công nghệ Bưu chính Viễn thông. Cuốn sách được xây dựng theo cấu trúc mô đun, thích hợp cho cả bạn đọc chỉ có nhu cầu tìm hiểu tổng quan tới các bạn đọc là các kỹ sư, cán bộ khoa học đang hoạt động trong lĩnh vực viễn thông có nhu cầu nghiên cứu chuyên sâu xử lý tín hiệu số (DSP). Với sự hỗ trợ mạnh mẽ của phần mềm Matlab, tác giả đã xây dựng nhiều hàm nhằm làm sáng tỏ các công thức toán học mang tới cái nhìn trực quan cho bạn đọc.

Cuốn sách được trình bày trong 18 chương với các nội dung cơ bản từ phương pháp luận mô phỏng, kỹ thuật mô phỏng tới các ví dụ mô phỏng hệ thống thực tế như hệ thống CDMA, FDM,... Cuốn sách còn có đĩa CD-ROM kèm theo, chứa các chương trình, hàm mô phỏng viết bằng ngôn ngữ Matlab cùng một số chuyên mục chi tiết dành cho bạn đọc cần tìm hiểu sâu hơn.

Hy vọng việc hiểu rõ các kỹ thuật, phương pháp mô phỏng khoa học sẽ hỗ trợ thiết thực cho chương trình nghiên cứu của bạn đọc đang công tác trong lĩnh vực viễn thông; các cán bộ khoa học, kỹ sư và sinh viên đang nghiên cứu, tìm hiểu và học tập

trong lĩnh vực viễn thông. Nhà xuất bản Bưu điện giới thiệu cùng bạn đọc và rất mong muốn nhận được những ý kiến góp ý để cuốn sách hoàn chỉnh hơn trong lần xuất bản sau. Mọi ý kiến xin gửi về Nhà xuất bản Bưu điện, 18 Nguyễn Du, Hà Nội; điện thoại: 04.9431283; Fax: 04.9431285.

Hà Nội, tháng 5 năm 2007

NHÀ XUẤT BẢN BƯU ĐIỆN

Chương 1
VAI TRÒ CỦA MÔ PHỎNG

1.1. Mở đầu

Mức độ phức tạp của các hệ thống truyền thông hiện đại là động lực để sử dụng mô phỏng. Tính phức tạp là do: *(i)* Cấu trúc phức tạp của hệ thống thông tin hiện đại; *(ii)* Môi trường trong đó các hệ thống này được triển khai. Yêu cầu các hệ thống truyền thông hiện đại hoạt động tốc độ cao với độ rộng băng tần hạn chế, công suất hạn chế. Các yêu cầu đối lập này dẫn đến định dạng xung, điều chế phức tạp cùng với mã hoá kiểm soát lỗi và tăng mức độ xử lý tín hiệu ở máy thu; *(iii)* Các yêu cầu về đồng bộ cũng trở nên chặt chẽ hơn tại tốc độ cao dẫn đến máy thu trở nên phức tạp hơn. Trong khi việc phân tích các hệ thống truyền thông tuyến tính làm việc trong môi trường kênh AWGN là đơn giản, thì hầu hết các hệ thống hiện đại làm việc trong các môi trường khắc nghiệt hơn. Các hệ thống nhiều chặng cần có các bộ khuếch đại *phi tuyến*. Các hệ thống vô tuyến tế bào thường làm việc trong môi trường nhiễu nghiêm trọng cùng với các ảnh hưởng của che chắn và đa đường gây thăng giáng tín hiệu thu. Vì vậy, các hệ thống phức tạp và các yêu cầu đối nghịch nhau dẫn đến bài toán thiết kế và phân tích không còn khả thi với các kỹ thuật truyền thống.

Sự phát triển của các máy tính số về khả năng xử lý, giá thành, tính thân thiện sử dụng... làm cho việc phân tích, thiết kế được trợ giúp bởi máy tính là rất hữu hiệu. Phát triển các gói phần mềm cho các hệ thống truyền thông hay viễn thông đã thúc đẩy việc sử dụng mô phỏng trong lĩnh vực này. Theo đó, gia tăng tính phức tạp của hệ thống đồng nghĩa với gia tăng mức độ tính toán. Trong nhiều trường hợp, khả năng tính toán phù hợp trực tiếp dẫn đến nhiều cấu trúc xử lý tín hiệu phức tạp mà tạo thành các khối chức năng của các hệ thống truyền thông hiện đại.

Tăng trưởng công nghệ máy tính cũng đồng nghĩa với tăng trưởng nhanh về lý thuyết mô phỏng. Kết quả, cần có các công cụ và các phương pháp luận để ứng dụng thành công mô phỏng cho các bài toán phân tích và thiết kế.

Động cơ thúc đẩy quan trọng dẫn đến dùng mô phỏng: *(i)* Là công cụ quý giá mang lại sự hiểu biết sâu sắc tính cách hệ thống; *(ii)* Triển khai mô phỏng phù hợp giống như thực thi một phòng thí nghiệm hệ thống; *(iii)* Dễ dàng đo kiểm tại các điểm khác nhau của hệ thống; *(iv)* Dễ dàng quản lý kiểm soát việc nghiên cứu tham số, vì các giá trị tham số như: độ rộng băng bộ lọc, SNR có thể bị thay đổi và sẽ ảnh hưởng lên hiệu năng hệ thống một cách nhanh chóng và quan sát được; *(v)* Dễ dàng tạo ra các dạng sóng trong miền thời gian và miền tần số như: phổ tín hiệu, biểu đồ mắt, hình sao tín hiệu cũng như các hiển thị khác; *(vi)* Dễ dàng so sánh và đánh giá các kết quả.

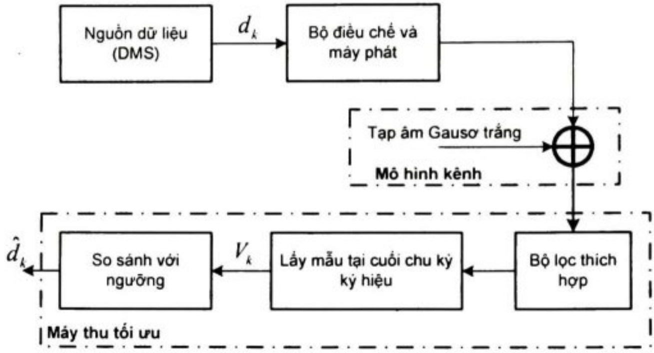
1.2. Minh họa mức độ phức tạp

Mức độ phức tạp của hệ thống truyền thông thay đổi rất rộng. Để được tường minh, ta xét ba hệ thống truyền thông với mức độ phức tạp tăng dần. Ta sẽ thấy rõ, hệ thống đầu tiên không cần thiết mô phỏng; hệ thống thứ hai việc mô phỏng không nhất thiết phải có nhưng nếu có là

hữu hiệu; hệ thống thứ ba, cần thiết phải thực hiện mô phỏng để kiểm soát quản lý nghiên cứu hiệu năng một cách chi tiết.

1.2.1. Hệ thống dễ xử lý theo phép giải tích

Một hệ thống truyền thông đơn giản nhất được minh họa ở hình 1.1.



Hình 1.1: Hệ thống truyền thông xử lý được theo phép giải tích

Nguồn dữ liệu (DMS): Khối nguồn dữ liệu tạo ra chuỗi các ký hiệu rời rạc d_k, được coi là các phần tử từ một thư viện ký hiệu hữu hạn. Chẳng hạn, với hệ thống truyền thông nhị phân gồm hai ký hiệu {0,1}. Ngoài ra, nguồn được coi là *không nhớ* nghĩa là ký hiệu thứ *k* được tạo ra từ nguồn độc lập với tất cả các ký hiệu khác được tạo ra từ nguồn đó. Nguồn dữ liệu thoả mãn hai tính chất này được gọi là nguồn không nhớ rời rạc DMS.

Bộ điều chế và máy phát: Vai trò của bộ điều chế là sắp xếp các ký hiệu nguồn thành các dạng sóng, mỗi dạng sóng thể hiện cho mỗi ký hiệu nguồn. Chẳng hạn, hệ thống truyền thông nhị phân có hai dạng sóng có thể được tạo ra từ bộ điều chế $\{s_1(t), s_2(t)\}$. Trường hợp này, máy phát được giả định chỉ khuếch đại tín hiệu đầu ra bộ điều chế sao cho các tín hiệu từ bộ điều chế được phát xạ với năng lượng mong muốn trên bit.

Mô hình kênh: Ở dạng tổng quát, việc mô hình hóa chính xác kênh vô tuyến là phần khó nhất của hệ thống. Tuy nhiên, ở đây ta đơn giản hoá, kênh chỉ là *cộng* tạp âm vào tín hiệu truyền qua nó. Tạp âm này cũng được giả thiết rằng có mật độ phổ công suất PSD không đổi trên toàn bộ dải tần. Tạp âm thoả mãn tính chất PSD không đổi được coi là tạp âm *trắng*. Biên độ tạp âm cũng được giả định là có hàm mật độ xác suất phân bố *Gausơ*. Kênh trong đó tạp âm là phân bố Gausơ, trắng, cộng được gọi là kênh AWGN. Nói cách khác kênh AWGN là kênh thỏa mã ba tính chất đề cập trên.

Máy thu tối ưu: Chức năng của máy thu là quan trắc tín hiệu vào, từ quan trắc này tạo ra một ước tính $\hat{d}_k$ của tín hiệu dữ liệu gốc d_k. Máy thu được minh họa ở hình 1.1 được xem là máy thu tối ưu vì việc thực hiện ước tính ký hiệu làm giảm thiểu xác suất lỗi P_E. Thấy rõ từ lý thuyết truyền thông số cơ bản, máy thu tối ưu cho hệ thống được mô tả ở trên (tín hiệu nhị phân

trong môi trường kênh AWGN) gồm một bộ lọc thích hợp (hoặc máy thu tương quan) thực hiện quan trắc tín hiệu trong một chu kỳ ký hiệu. Đầu ra bộ lọc thích hợp được lấy mẫu tại thời điểm cuối của chu kỳ ký hiệu để tạo ra giá trị V_k. V_k là một biến ngẫu nhiên (vì kênh tác động vào tín hiệu truyền qua nó ở dạng toán tử cộng, phân bố Gausơ, mọi tần số) và được so sánh với ngưỡng T. Nếu $V_k > T$, thì quyết định là ký hiệu 1, ngược lại quyết định là ký hiệu 0.

Ta coi hệ thống này là hệ thống xử lý được theo phép giải tích vì theo lý thuyết truyền thông cơ bản, việc phân tích hệ thống được thực hiện dễ dàng. Chẳng hạn, xác suất lỗi được tìm thấy là:

$$P_E = Q\left(\sqrt{k\frac{E_s}{N_0}}\right) \tag{1.1}$$

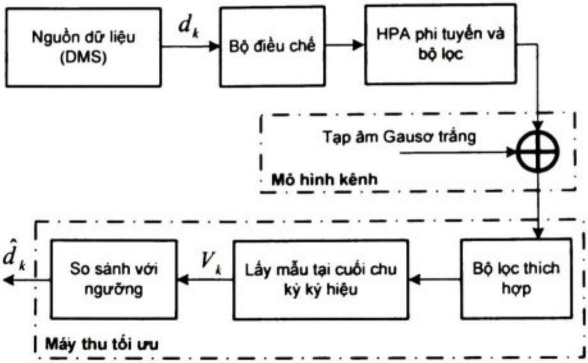
Trong đó E_S thể hiện năng lượng trung bình được tính toán trong chu kỳ ký hiệu tương ứng với tập các dạng sóng $\{s_1(t), s_2(t)\}$, và N_0 là mật độ phổ công suất một phía của tạp âm kênh cộng. Tham số k được xác định bởi tương quan của các dạng sóng $\{s_1(t), s_2(t)\}$. Ví dụ: với truyền dẫn FSK, thì dạng sóng $\{s_1(t), s_2(t)\}$ là các hình **sin** có tần số khác nhau và công suất bằng nhau. Giả sử tần số được chọn chính xác thì các tín hiệu này không tương quan nhau và $k = 1$. Đối với PSK, thì các tín hiệu được dùng để truyền dẫn dữ liệu được coi là hình sin có tần số và công suất bằng nhau nhưng các pha ban đầu khác nhau. Nếu sự khác nhau về pha là π radian sao cho $s_2(t) = -s_1(t)$ thì các tín hiệu đối tương quan và $k = 2$.

Hiệu năng hệ thống: Dễ dàng xác định được hiệu năng của hệ thống được cho ở hình 1.1 bằng các kỹ thuật phân tích truyền thống. Vì vậy, ta có thể phân loại hệ thống thuộc loại hệ thống xử lý được theo phép giải tích. *Tại sao hệ thống này xử lý được theo phép giải tích?* Vì 03 lý do cơ bản sau: *(i)* Kênh AWGN và máy thu tuyến tính. Theo đó, vì tạp âm là Gausơ và bộ lọc thích hợp là hệ thống tuyến tính, nên giá trị quyết định V_k là một biến ngẫu nhiên Gausơ vì vậy ta có thể tính BER theo phép giải tích là một hàm của các tham số bộ lọc máy thu, việc xác định các giá trị của các tham số này dẫn đến BER cực tiểu. Nhiều nhân tố dẫn đến hệ thống được cho ở hình 1.1 là hệ thống xử lý được theo phép giải tích. Các nhân tố này gắn liền việc đơn giản hoá trong quá trình lập mô hình hệ thống (do các giả định hay điều kiện xét...); *(ii)* Nguồn dữ liệu là không nhớ (thực tế nó có thể là đúng hoặc không đúng); *(iii)* Giả thiết việc đồng bộ ký hiệu chính xác vì vậy biết chính xác thời điểm bắt đầu và kết thúc của các ký hiệu dữ liệu. Giả định này cho phép giá trị quyết định V_k được trích ra một cách chính xác.

Vậy mô phỏng có quan trọng trong hệ thống xử lý được theo phép giải tích không? Câu trả lời là có. Vì hệ thống được cho ở hình 1.1 sẽ là một khối cơ bản của hệ thống phức tạp hơn. Mã chương trình mô phỏng có thể được phát triển, nâng cấp cho hệ thống phức tạp hơn. Thành quả mô phỏng dễ dàng được thừa nhận vì việc phân tích hệ thống đó là dễ hiểu. Tại đây, theo yêu cầu của hệ thống trong điều kiện nghiên cứu cụ thể, để mô hình hóa hệ thống một cách chính xác cần phải biến đổi các khối nguồn dữ liệu, bộ điều chế, kênh, máy thu cho phù hợp. Ngoài ra, các phân hệ (hệ thống con) khác khi cần có thể được đưa thêm vào mô hình mô phỏng đó. Do tiếp tục nhiệm vụ phát triển mô hình mô phỏng từ hệ thống đó, nên có thể tin tưởng rằng điểm bắt đầu là đúng.

1.2.2. Hệ thống khó xử lý theo phép giải tích

Xét hệ thống ở mức độ phức tạp hơn. Theo đó, ta mở rộng mức độ phức tạp hệ thống hình 1.1 thành hệ thống được cho ở hình 1.2 bằng cách thêm khối khuếch đại công suất cao phi tuyến HPA và bộ lọc ở máy phát.



Hình 1.2: Hệ thống truyền thông khó xử lý theo phép giải tích

Bộ khuếch đại phi tuyến: Hiệu quả công suất của các bộ khuếch đại phi tuyến cao hơn nhiều so với bộ khuếch đại tuyến tính vì vậy thường được dùng trong môi trường công suất bị hạn chế như: các ứng dụng không gian vũ trụ, các hệ thống thông tin di động ở đó công suất pin phải được duy trì. Do tính phi tuyến gây ra méo điều chế ký sinh và tạo hài. Hậu quả làm nở rộng phổ tín hiệu vào trong khi đó bộ khuếch đại tuyến tính bảo tồn phổ tín hiệu vào. Bộ lọc làm giảm bớt hài và méo điều chế ký sinh (do tính phi tuyến gây ra) nhưng lại gây ra *tán thời* tín hiệu. Theo đó, tín hiệu được lọc này không còn bị giới hạn về thời gian so với chu kỳ ký hiệu nữa, dẫn đến giao thoa giữa các ký hiệu ISI.

Hiệu năng BER hệ thống: Hậu quả của ISI làm cho xác suất lỗi của ký hiệu thứ *i* phụ thuộc vào một hoặc nhiều ký hiệu trước đó. Số lượng các ký hiệu trước đó phải được tính đến trong quá trình giải điều chế ký hiệu thứ *i* này (lưu ý tính có nhớ của hệ thống). Cũng vậy, nếu xác suất lỗi ký hiệu thứ *i* phụ thuộc vào *k* ký hiệu trước đó thì ta tính đại lượng:

$$\Pr\left\{E_i \middle| d_{i-1}d_{i-2}...d_{i-k}\right\}$$

Trường hợp nhị phân, có 2^k *chuỗi khác nhau có độ dài k. Nếu xác suất xuất hiện mỗi ký hiệu dữ liệu {0,1} bằng nhau thì xác suất lỗi của ký hiệu thứ i là:*

$$P_E = \frac{1}{2^k}\sum_{d_{i-1}=0}^{1}\sum_{d_{i-2}=0}^{1}...\sum_{d_{i-k}=0}^{1}\Pr\left\{E_i \middle| d_{i-1}d_{i-2}...d_{i-k}\right\} \quad (1.2)$$

Nói cách khác, cần phải tính 2^k xác suất lỗi khác nhau, với mỗi xác suất lỗi phụ thuộc vào một trong 2^k chuỗi độ dài *k* trước đó và lấy trung bình *k* kết quả. Vì kênh được giả định là

kênh AWGN, nên mỗi xác suất lỗi trong 2^k xác suất lỗi này là một hàm Q Gausơ. Tuy dễ hiểu, nhưng khó tính toán đối số của mỗi hàm Q vì vậy thường dùng mô phỏng.

Hệ thống được minh họa ở hình 1.2 có tính chất quan trọng là dễ phân tích. Lưu ý rằng hệ thống là tuyến tính do kênh AWGN (hệ thống là tuyến tính được xét từ điểm tạp âm được phun vào tới điểm xuất hiện V_k). Vì vậy, con số thống kê V_k thường có dạng:

$$V_k = S_k + I_k + N_k \tag{1.3}$$

Trong đó I_k , S_k là thành phần do giao thoa giữa các ký hiệu và tín hiệu mang tin, và N_k là thành phần do tạp âm kênh. Vì vậy, nếu tạp âm kênh là Gausơ thì N_k là biến ngẫu nhiên Gausơ (lưu ý chuyển đổi tuyến tính của biến ngẫu nhiên Gausơ). Ngoài ra, V_k là biến ngẫu nhiên Gausơ có cùng phương sai với N_k nhưng giá trị trung bình là $S_k + I_k$ (cả hai đều là tất định). Có thể dễ dàng tìm được giá trị trung bình của V_k. Phương sai của V_k được xác định từ mật độ phổ công suất của kênh tạp âm và độ rộng băng thông tạp âm tương đương. Biết hàm mật độ xác suất pdf của V_k, dễ dàng tìm được xác suất lỗi. Tóm lại, dễ dàng tìm được hàm mật độ xác suất pdf của V_k ngay cả khi hệ thống là phi tuyến vì tạp âm không qua phần tử phi tuyến của hệ thống (chỉ có HPA là phi tuyến và tạp âm do kênh).

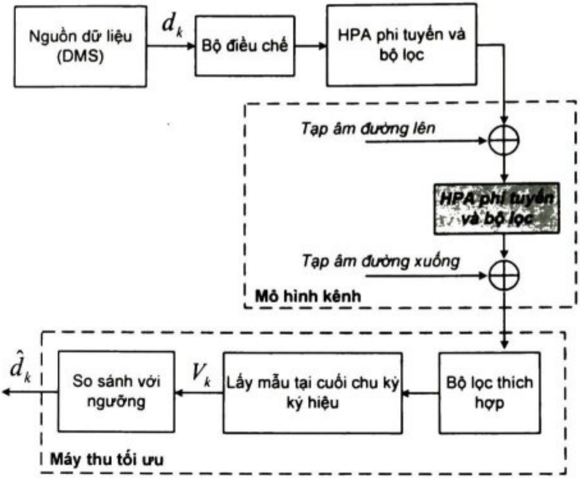
Trường hợp cụ thể này, do tạp âm chỉ đi qua phần tuyến tính của hệ thống, nên có ảnh hưởng quan trọng đến phương pháp luận mô phỏng. Vì tạp âm không qua phần tử phi tuyến, nên nhanh chóng xác định được trung bình của V_k bằng mô phỏng phi tạp âm. Phương sai của V_k có thể xác định theo phép giải tích, kết quả là biết được pdf của V_k và dễ dàng xác định được xác suất lỗi. Các khái niệm này được kết hợp trong kỹ thuật mô phỏng một cách nhanh chóng và đơn giản. Kết quả là phương pháp bán giải tích (nghĩa là việc phân tích và mô phỏng được kết hợp với nhau), dẫn đến mô phỏng rất nhanh, vì vậy sẽ là công cụ rất quan trọng và là chủ đề của các chương sau.

1.2.3. Hệ thống không thể xử lý theo phép giải tích

Hệ thống được minh họa ở hình 1.3 được coi là hệ thống không thể xử lý theo phép giải tích, là mô hình đơn giản của hệ thống thông tin vệ tinh hai chặng. Bộ phát đáp vệ tinh được mô hình hóa như là bộ khuếch đại công suất cao HPA phi tuyến và bộ lọc thực hiện khử méo hài ngoài băng do tính phi tuyến của HPA gây ra. So sánh hình 1.2 và hình 1.3, chúng khá giống nhau. Trường hợp này, mô hình kênh vệ tinh bao gồm hai nguồn tạp âm (tạp âm đường lên và tạp âm đường xuống). *Vấn đề* là tạp âm ở máy thu gồm tạp âm đường xuống, và tạp âm đường lên đi qua bộ HPA *phi tuyến*. Thậm chí coi rằng cả tạp âm đường lên và đường xuống đều là phân bố Gausơ, nhưng vẫn rất khó xác định hàm pdf của tạp âm máy thu. Từ hình vẽ cho thấy, dễ dàng mô hình hóa tạp âm đường xuống vì nó chỉ qua phần *tuyến tính* của hệ thống nhưng tạp âm đường lên rất khó xác định vì nó phải qua HPA *phi tuyến*. Dù là tạp âm đường lên là phân bố Gausơ nhưng pdf của nó tại đầu vào máy thu không còn là phân bố Gausơ nữa. Theo đó, việc xác định pdf của Vk là rất khó. Vì vậy, mô phỏng là công cụ thiết yếu cho các loại hệ thống này.

Phạm vi các hệ thống được xét trong phần này là rất hẹp. Các hệ thống được chọn chỉ nhằm mục đích minh họa: tăng mức độ phức tạp dẫn đến mô phỏng. Rất nhiều hệ thống hiện nay thuộc loại hệ thống không thể xử lý được bằng giải tích. Ví dụ các hệ thống thông tin di

động làm việc trong môi trường đa đường và nhiễu cao. Việc mô phỏng luôn cần thiết để phân tích chi tiết các hệ thống như vậy.



Hình 1.3: Hệ thống truyền thông không thể xử lý theo phép giải tích

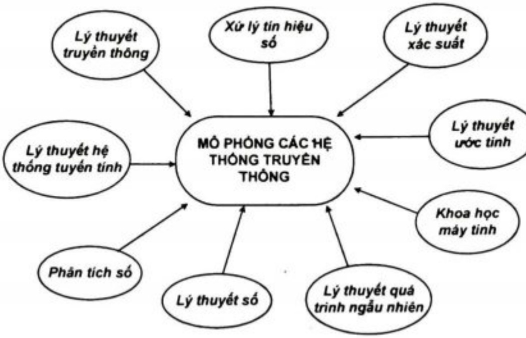
1.3. Các mặt đa kỷ luật của mô phỏng

Trước những năm 1970 bài toán mô phỏng thường được giải quyết theo cách thiên về dạng đặc biệt. Phương pháp luận để phát triển mô phỏng và các nguồn lỗi xuất hiện trong mọi chương trình mô phỏng không được hiểu một cách đầy đủ. Hơn 20 năm qua, cộng đồng nghiên cứu đã tạo ra một khối lượng lớn kiến thức, tạo ra phương pháp luận để phát triển mô phỏng cũng như việc thống nhất về lý thuyết để giải quyết nhiều vấn đề nảy sinh trong quá trình triển khai chương trình mô phỏng. Theo đó, việc dùng mô phỏng như là công cụ giải tích cần thiết để hiểu và hiểu sâu sắc nhằm triển khai mô phỏng khả tin. Cần phải xây dựng khối lượng lớn kiến thức tích hợp từ nhiều tài liệu trong nhiều lĩnh vực khác nhau. Mặc dù chưa được thấu đáo nhưng 09 lĩnh vực nghiên cứu quan trọng ảnh hưởng đến quá trình nghiên cứu về mô phỏng được mô tả ở hình 1.4. Ta xét ngắn gọn 09 lĩnh vực này nhằm rõ hơn về mối quan hệ của chúng với khoa học mô phỏng.

Các khái niệm về *lý thuyết hệ thống tuyến tính* cho ta các kỹ thuật để xác định các quan hệ vào/ra của hệ thống tuyến tính, cho phép trình bày mô hình hệ thống trong miền thời gian ở dạng hàm đáp ứng xung kim hệ thống và miền tần số ở dạng hàm truyền đạt hệ thống cũng như việc xây dựng nền tảng cho nhiều vấn đề.

Hiển nhiên, kiến thức *lý thuyết truyền thông* là rất quan trọng. Cấu trúc hệ thống, đặc tính hoạt động của các phân hệ (bộ giải điều chế, bộ cân bằng, chi tiết hóa các mô hình kênh...) phải được hiểu rõ trước khi triển khai mô phỏng. Khi sử dụng mô phỏng để xác định các giá trị của

tham số hệ thống, cần phải lưu ý đến dải giá trị của nó có ý nghĩa thực tế trước khi triển khai mô phỏng. Cần phải có những hiểu biết sâu sắc về tính cách hệ thống để đảm bảo hoạt động mô phỏng chính xác và kết quả hợp lý.



Hình 1.4: Các lĩnh vực ảnh hưởng lên nghiên cứu mô phỏng các hệ thống truyền thông

Các công cụ của *xử lý tín hiệu số* (DSP) được dùng để triển khai các giải thuật, từ đó xây dựng mô hình mô phỏng hệ thống truyền thông. Mô hình mô phỏng này thường bao gồm một số phép lấy xấp xỉ rời rạc của các phần tử hệ thống liên tục, cần có kiến thức về xử lý tín hiệu số để hiểu và đánh giá bản chất của các phép lấy xấp xỉ này. Thực tế, mỗi khối chức năng trong mô hình mô phỏng là một hoạt động DSP, vì vậy các công cụ của DSP cho ta các kỹ thuật thực hiện mô phỏng.

Giải tích số có quan hệ chặt chẽ với DSP, nhưng lưu ý tách biệt vì nó là phần kiến thức cũ hơn. Nhiều kỹ thuật kinh điển như tích phân số, nội suy đa thức, phù hợp hóa đồ thị đều có nguồn gốc trong giải tích số.

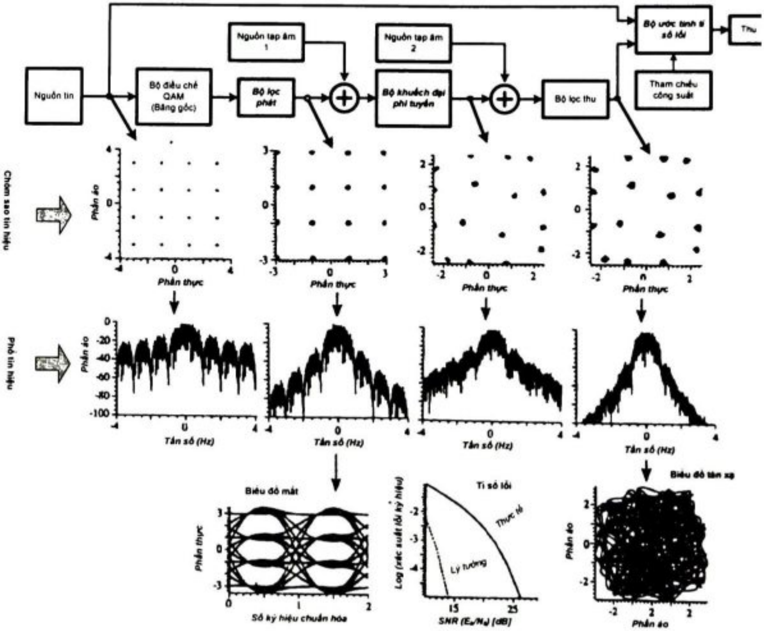
Các khái niệm về *xác suất* cũng là nền tảng căn bản cho nghiên cứu. Việc đánh giá hiệu năng hệ thống truyền thông thường được biểu diễn trong các thuật ngữ xác suất. Chẳng hạn: khi đề cập xác suất lỗi bit hay xác suất lỗi ký hiệu trong hệ thống truyền thông số; khi xét bài toán đồng bộ, ta quan tâm xác suất lỗi pha vượt quá một mức cho trước. Lý thuyết xác suất cơ bản cho ta khái niệm về biến ngẫu nhiên và hàm mật độ xác suất. Kiến thức về hàm mật độ xác suất cơ bản cho phép ta tính toán các đại lượng như đã đề cập phần trên. Phần sau ta sẽ trình bày rõ, kết quả của nhiều mô phỏng (được gọi là các mô phỏng ngẫu nhiên (stochastic)) là biến ngẫu nhiên điển hình và phương sai của nó thường đánh giá tính hữu hiệu và tính chính xác thống kê của mô phỏng.

Trong nhiều trường hợp, dạng sóng tín hiệu và tạp âm được xử lý bởi mô phỏng được coi là các hàm mẫu của một quá trình ngẫu nhiên. Việc triển khai các thuật toán để tạo dạng sóng có các thuộc tính thống kê phù hợp cần có kiến thức quá trình ngẫu nhiên cơ bản, nó đặc biệt có ý nghĩa khi triển khai các mô hình mô phỏng kênh. Lý thuyết quá trình ngẫu nhiên cho ta các công cụ để mô tả các quá trình này trong miền thời gian (ví dụ hàm tự tương quan), và

trong miền tần số (ví dụ mật độ phổ công suất). Ta cũng đề cập nhiều ứng dụng khác của thuyết quá trình ngẫu nhiên.

Cần hiểu sơ lược khái niệm *lý thuyết số*, nó cho ta các công cụ để triển khai các bộ tạo ngẫu nhiên. Các bộ tạo số ngẫu nhiên này là các khối cơ bản của bộ tạo dạng sóng để biểu d các chuỗi số, dạng sóng tạp âm, pha đinh tín hiệu, nhiễu ngẫu nhiên...

Khái niệm cơ bản về *khoa học máy tính* cũng sẽ hữu hiệu. Ví dụ: độ dài từ mã, khu dạng từ mã được dùng để biểu diễn các mẫu tín hiệu, sẽ ảnh hưởng đến tính chính xác của phỏng. Việc chọn ngôn ngữ cũng quan trọng khi triển khai các bộ mô phỏng thương mại. nhớ khả dụng, tổ chức nhớ sẽ ảnh hưởng cách thức dữ liệu và lệnh được qua các phần của phỏng. Các yêu cầu về đồ họa và dung lượng sẽ xác định dạng sóng được hiển thị như thế r và sẽ ảnh hưởng quá trình truyền tải mã chương trình mô phỏng từ máy tính này đến máy tính kh



Hình 1.5: Minh họa mô phỏng mức dạng sóng cho một hệ thống truyền thông điển hình

Các công cụ và các khái niệm về *lý thuyết ước tính* cho phép định lượng tính hiệu quả c kết quả mô phỏng. Kết quả mô phỏng ngẫu nhiên là một biến ngẫu nhiên. Mỗi khi thực h mô phỏng sẽ tạo ra một *giá trị* của biến ngẫu nhiên đó (lưu ý: biến ngẫu nhiên giá trị thực một hàm của giá trị thực), biến ngẫu nhiên này tạo thành *bộ ước tính* cho đại lượng cần đu ước tính. Một cách điển hình, tất cả các giá trị được tạo ra bởi bản sao của *mô phỏng* sẽ kh

nhau. Việc mô phỏng là hữu hiệu nhất khi bộ ước tính (nó được tạo ra bởi mô phỏng) là *không chệch và kiên định*. *Ước tính không chệch* là ước tính trong đó giá trị trung bình của ước tính là một đại lượng đo (đại lượng đo kiểm thực tế). Đây là cách nói khác về việc trung bình hóa các ước tính được tạo ra bởi mô phỏng là chính xác. Rõ ràng đây là một thuộc tính mong muốn. *Ước tính kiên định* là ước tính mà phương sai của nó giảm khi thời gian mô phỏng tăng. Nói cách khác, nếu thực hiện đo 100 lần độc lập về độ cao của một người và lấy trung bình kết quả, ta mong đợi kết quả chính xác hơn so với trường hợp đo một lần. Tổng quát, lý thuyết ước tính cho ta các công cụ giải tích cần thiết để đánh giá mức độ khả tin của các kết quả mô phỏng.

Tóm tắt ước tính tỉ số lỗi thông qua mô phỏng bao gồm các bước cơ bản sau:

- Tạo các giá trị mẫu của quá trình đầu vào (dạng sóng).

- Xử lý các mẫu thông qua mô hình và tạo các mẫu đầu ra.

- Ước tính tỉ số lỗi bằng cách so sánh các giá trị được mô phỏng của chuỗi đầu vào và dạng sóng đầu ra.

Một minh họa cho mô phỏng mức dạng sóng được cho ở hình 1.5, giúp trực quan hóa các các tín hiệu được mô phỏng tại các điểm đặc trưng của hệ thống.

Các vấn đề được tóm lược ở trên không có hàm ý làm cho việc nghiên cứu mô phỏng trở nên nặng nề mà đơn giản chỉ để thấy rằng việc mô phỏng là một lĩnh vực nghiên cứu chính đáng. Nó cuốn hút các lĩnh vực khác.

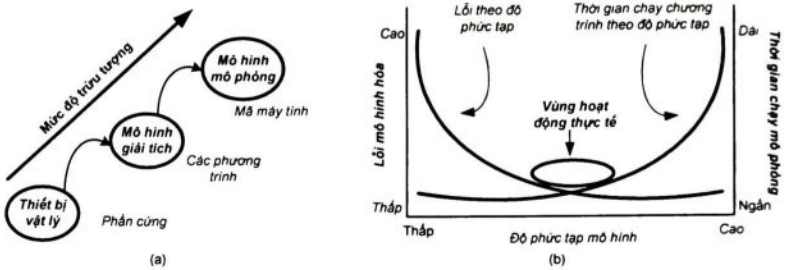
1.4. Các mô hình

Bước đầu tiên trong quá trình triển khai mô phỏng hệ thống truyền thông là triển khai mô hình mô phỏng cho hệ thống đó hay lập mô hình. Ta làm quen với mô hình và nên hiểu rằng mô hình mô tả quan hệ vào/ra của thiết bị hay hệ thống vật lý. Cụ thể, các mô hình được biểu biễn ở dạng toán học. *Nghệ thuật* lập mô hình là triển khai các mô hình tính cách (ta dùng thuật ngữ này vì mô hình đó bắt giữ tính cách của thiết bị trong điền kiện cụ thể), đủ mức độ chi tiết để duy trì các nét đặc trưng chủ yếu của của hệ thống và không quá phức tạp sao cho mức độ chi phí tài nguyên tính toán hợp lý. Vì vậy, cần phải dung hòa các tiêu chí đối lập như độ chính xác, mức độ phức tạp, yêu cầu tính toán.

Cần phải nghiên cứu hai loại mô hình đó là: ***mô hình giải tích và mô hình mô phỏng***. Cả hai đều là sự trừu tượng hóa của thiết bị vật lý hoặc hệ thống như được minh họa ở hình 1.6(a). Thiết bị trong hình 1.6(a) có thể chỉ là một mạch điện tử hay phân hệ như vòng khóa pha PLL... Bước đầu tiên và bước quan trọng nhất trong quá trình lập mô hình là nhận biết các thuộc tính và các đặc tính hoạt động của thiết bị sẽ được trình bày trong mô hình. Việc nhận biết các tính năng cốt lõi này thường cần có các phán đoán thiết kế đáng kể và luôn luôn cần phải có hiểu biết thấu đáo về ứng dụng mà mô hình đó sẽ được triển khai. Mức độ chính xác cần thiết của bất kỳ mô phỏng hay phân tích toán học nào dựa trên mô hình đó đều bị hạn chế bởi tính chính xác của mô hình. Một khi các vấn đề này được giải quyết, thì ta triển khai mô hình giải tích để nắm giữ các tính năng cốt lõi của thiết bị vật lý. Một cách điển hình, mô hình giải tích có dạng phương trình hoặc hệ phương trình để xác định quan hệ vào/ra của thiết bị. Các phương trình này chỉ là sự mô tả từng phần của thiết bị được mô hình hóa vì chỉ có một khía cạnh nào đó của thiết bị được mô hình hóa. Ngoài ra, các phương trình định nghĩa cho thiết bị đó chỉ chính xác

trong một phạm vi giới hạn của điện áp, dòng điện hay tần số. Mô hình mô phỏng thường là một tập hợp các giải thuật nhằm tìm nghiệm số của các phương trình (các phương trình này đang định nghĩa mô hình giải tích đó). Các kỹ thuật phân tích số và xử lý tín hiệu số là các công cụ để triển khai các giải thuật này.

Thấy rõ từ hình 1.6(a), mức độ trừu tượng tăng dần từ thiết bị qua mô hình giải tích cuối cùng là mô hình mô phỏng. Việc tăng mức độ trừu tượng là do giả định và xấp xỉ hóa trong quá trình di chuyển từ thiết bị vật lý - mô hình giải tích - mô hình mô phỏng.



Hình 1.6: (a) Các thiết bị và các mô hình; (b) Các ảnh hưởng của tính phức tạp mô hình

Trong nghiên cứu ta thường gặp phải các mô hình có các mức độ trừu tượng khác nhau. Ví dụ: ta sẽ thấy rằng, các kênh có thể được mô hình hóa bằng giải pháp mức dạng sóng trong đó các giá trị mẫu của các dạng sóng được xử lý bởi mô hình đó. Mặt khác, các kênh có thể được trình bày bởi một quá trình Markov rời rạc dựa trên các ký hiệu chứ không phải dựa trên các mẫu của các dạng sóng. Ngoài ra, các mô hình Markov thường lôi cuốn bộ điều chế, máy phát, máy thu vào kênh. Các mô hình này khá trừu tượng và khó khăn khi tham số hóa một cách chính xác nhưng một khi tìm được thì mang lại các mô phỏng hiệu quả cao và thực hiện nhanh. Tính hiệu quả này là lý do chính dẫn đến mô hình hóa ở mức trừu tượng hơn.

Hình 1.6(b) thấy rõ hơn về quá trình mô hình hóa. Trực quan mà nói, thuộc tính mong muốn của mô phỏng là thực hiện mã mô phỏng nhanh. Các mô hình đơn giản thực hiện nhanh hơn các mô hình phức tạp vì cần ít đường nối đến mã máy tính hơn mỗi khi mô hình được thể hiện bởi mô phỏng đó. Tuy nhiên, các mô hình đơn giản không đặc trưng hóa hết được các thuộc tính quan trọng của thiết bị dẫn đến các kết quả không chính xác, khi này cần có các mô hình phức tạp hơn song lại tăng thời gian mô phỏng.

1.5. Mô phỏng tất định và mô phỏng ngẫu nhiên

Tồn tại hai loại mô phỏng cơ bản là mô phỏng tất định và mô phỏng ngẫu nhiên.

Mô phỏng tất định: Loại này hầu như quen thuộc với ta. Ví dụ mô phỏng mạch điện tử cố định, trong đó quan tâm đáp ứng cho tín hiệu vào tất định cụ thể. Triển khai chương trình phần mềm để biểu diễn thành phần mạch và tín hiệu vào. Mô phỏng tạo ra dòng điện, điện áp tương ứng trên mỗi nhánh, và thường được biểu diễn ở dạng sóng. Cần phải quy định khoảng thời gian biểu diễn dạng sóng trước khi mô phỏng. *Do tính bất biến của mạch và tính tất định*

của tín hiệu vào nên các kết quả mô phỏng như nhau sau mỗi lần mô phỏng. Ngoài ra, nhận được cùng dạng sóng nếu xử lý mạng bằng các kỹ thuật truyền thống. Mô phỏng sẽ tiết kiệm thời gian và tránh những lỗi toán học do tính toán dài dòng.

Mô phỏng ngẫu nhiên: Giả sử đưa vào mạng là dạng sóng ngẫu nhiên (hàm mẫu của quá trình ngẫu nhiên). Một cách tương đương, mô hình hệ thống cần có trở kháng của điện trở là một biến ngẫu nhiên được xác định bởi hàm mật độ xác suất cụ thể. Kết quả mô phỏng sẽ không còn là dạng sóng tất định nữa, và các mẫu của dạng sóng này sẽ tạo ra một tập các biến ngẫu nhiên. Những mô phỏng trong đó xuất hiện các đại lượng ngẫu nhiên được được quy vào mô phỏng ngẫu nhiên.

Ví dụ: Giả sử điện áp $e(t)$ trên một phần tử mạch và thực hiện mô phỏng tạo ra giá trị của $e(t)$ tại thời điểm 1 *ms*, nghĩa là muốn có $e(0{,}001)$. Trong mô phỏng tất định thì $e(0{,}001)$ là cố định và mỗi lần mô phỏng hay mỗi khi dùng kỹ thuật phân tích truyền thống đều cho cùng kết quả. Trong mô phỏng ngẫu nhiên thì $e(0{,}001)$ là một biến ngẫu nhiên và mỗi lần mô phỏng là một kết quả khác.

Ví dụ: Hệ thống truyền thông số trong đó tín hiệu thu gồm tín hiệu phát cộng với tạp âm ngẫu nhiên. Giả sử phải tính xác suất lỗi ký hiệu tại đầu ra máy thu. Thấy rõ, nếu truyền tín hiệu BPSK trong môi trường kênh AWGN thì xác suất lỗi ký hiệu là:

$$P_E = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{1.4}$$

Trong đó E_b là năng lượng ký hiệu, N_0 là mật độ phổ công suất tạp âm một phía và $Q(.)$ là hàm Q của Gausơ được định nghĩa là:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{y^2}{2}} dy \tag{1.5}$$

Lưu ý rằng P_E là một số chứ không phải là một biến ngẫu nhiên, mặc dù đầu vào máy thu là một đại lượng ngẫu nhiên. Số P_E là một trung bình trên vô hạn các thử nghiệm, mỗi thử nghiệm là một ký hiệu được qua hệ thống và quan sát kết quả. Tất nhiên kết quả là một quyết định đúng hoặc một lỗi được quan sát tại đầu ra máy thu. Đối với các quá trình ergodic có thể xác định xác suất lỗi theo hai cách: *(i)* Quan sát một bit đang được phát và tính P_E là *trung bình toàn bộ* trong đó ta có toàn bộ vô hạn các dạng sóng tạp âm có cùng các đặc tính thống kê; *(ii)* Xác định P_E là trung bình theo thời gian bằng cách phát nhiều vô hạn ký hiệu nhị phân và dùng *một hàm mẫu tạp âm*. Vấn đề then chốt là tính P_E dùng vô hạn ký hiệu phát. *Nếu* thay vì xác định P_E dựa vào vô hạn ký hiệu phát, ta *ước tính* P_E sử dụng hữu hạn ký hiệu nhị phân phát *thì* ta sẽ thấy rằng việc ước tính P_E thực ra là một biến ngẫu nhiên, vì mỗi hàm mẫu trong khoảng thời gian hữu hạn sẽ tạo ra một giá trị khác (hy vọng không quá khác) đối với xác suất lỗi. Điều này sẽ được sáng tỏ trong phần sau khi ta xét kỹ thuật Monte Carlo.

Cần lưu ý rằng: Trường hợp tất định, mô phỏng tất định và phân tích dẫn đến một con số. Mỗi khi thực hiện phân tích và mô phỏng sẽ nhận được cùng giá trị. Trường hợp ngẫu nhiên,

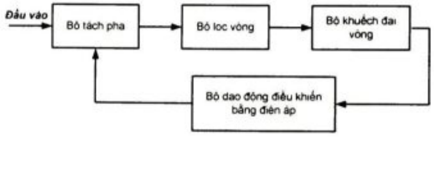
các mô phỏng ngẫu nhiên dẫn đến các biến ngẫu nhiên và tính cách thống kê của các biến ngẫu nhiên này là rất quan trọng trong việc xác định chất lượng các kết quả mô phỏng.

1.5.1. Minh họa mô phỏng tất định

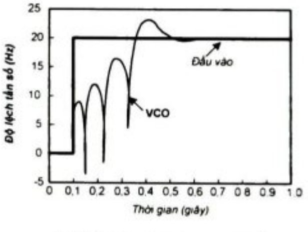
Mặc dù mục đích chính là trình bày và khám phá các kỹ thuật được sử dụng trong mô phỏng ngẫu nhiên nhưng cần lưu ý rằng mô phỏng tất định đầy đủ là những công cụ quan trọng để hiểu biết sâu sắc tính cách hoạt động của các hệ thống truyền thông. Mô phỏng để xác định dạng sóng tại các điểm quan tâm trong hệ thống. Các tham số hệ thống có thể bị thay đổi và dễ dàng khảo sát các ảnh hưởng. Có thể dùng các mô hình rất đơn giản mà vẫn có được các kết quả quan trọng.

Như ví dụ đơn giản: Xét vòng khóa pha PLL được dùng trong quá trình đồng bộ và giải điều chế được minh họa ở hình 1.7, nó khá đơn giản. Tuy nhiên, do các đặc tính *phi tuyến* của bộ tách pha nên việc phân tích vòng khóa pha trong *chế độ bắt* là khá phức tạp. Trường hợp đơn giản, tham số hiệu năng quan trọng của PLL là thời gian cần thiết để bắt được tín hiệu khi cho trước các tham số của PLL và các đặc tính kỹ thuật của tín hiệu vào. Để giải bài toán này theo cách giải tích, phải có nghiệm phương trình vi phân *phi tuyến*. Theo đó, cần chuyển hướng sang mô phỏng.

Giả sử PLL được thiết kế với tần số tự nhiên *(natural frequency)* là 5 Hz, và hệ số tắt dần *(damping factor)* theo thời gian là 0,707. Cũng coi rằng PLL đang hoạt động ở *chế độ khóa* và tần số đầu vào thay đổi tức thì 20 Hz tại $t = 0,1s$. Khi tỉ số giữa thay đổi bước nhảy tần số đầu vào với tần số tự nhiên của PLL là lớn, thì PLL sẽ mất khóa pha và phải bắt lại tín hiệu vào. Do tính cách phi tuyến của vòng dẫn đến hiện tượng gọi là "*trượt chu kỳ - cycle slipping*", và thời gian bắt giữ sẽ phụ thuộc nhiều vào số chu kỳ bị trượt trong quá trình bắt giữ.



Hình 1.7: Mô hình vòng khóa pha PLL



Hình 1.8: Tính cách bắt của vòng khóa pha PLL

Kết quả của một mô phỏng đơn giản được minh họa ở hình 1.8, trong đó bước nhảy tần số đầu vào xảy ra tại $t = 0,1s$. Ta thấy rằng PLL trượt 3 chu kỳ và sau đó bắt lại tại $t \approx 0,6s$. Mô phỏng là hoàn toàn tất định, và thực hiện nhiều lần mô phỏng sử dụng cùng các tham số của PLL và mô hình tín hiệu sẽ nhận được kết quả như nhau. Bài toán này sẽ được khai thác sâu hơn ở chương sau để nghiên cứu các kỹ thuật triển khai các mô phỏng hệ thống mà không phải chịu sự phức tạp do các dao động ngẫu nhiên.

1.5.2. Minh họa mô phỏng ngẫu nhiên

Tại đây ta xét một tình huống hoàn toàn khác. Xét hệ thống truyền thông số đơn giản được minh họa ở hình 1.1, và giả sử ta muốn tìm BER. Kỹ thuật mô phỏng cơ bản nhất để xác định phép đo hiệu năng quan trọng BER này là cho nhiều ký hiệu số qua hệ thống và đếm lỗi tại đầu ra máy thu, được gọi là kỹ thuật Monte Carlo. Nếu N ký hiệu được xử lý bởi hệ thống và quan trắc được N_e lỗi tại đầu ra máy thu thì ước tính Monte Carlo của xác suất lỗi là:

$$\hat{P}_E = \frac{N_e}{N} \tag{1.6}$$

Được biết là BER dựa trên N ký hiệu, và giá trị của BER tạo ra ước tính xác suất lỗi ký hiệu, sử dụng *định nghĩa tần xuất tương đối của xác suất*, ta có:

ĐẠI HỌC THÁI NGUYÊN
TRUNG TÂM HỌC LIỆU

$$P_E = \lim_{N \to \infty} \frac{N_e}{N} \tag{1.7}$$

Vì mô phỏng chỉ xử lý hữu hạn các ký hiệu nên xác suất lỗi ký hiệu chỉ được lấy xấp xỉ.

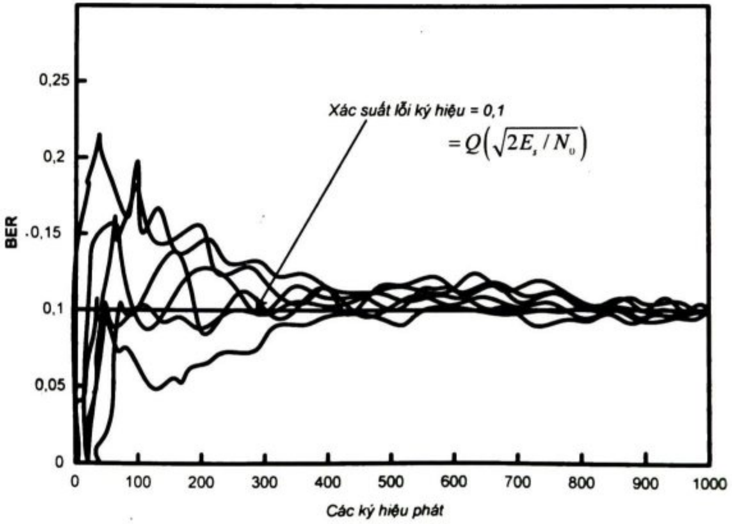
Bảng 1.1: So sánh sự khác nhau giữa BER và xác suất lỗi P_E

Tỉ số lỗi bit BER	**Xác suất lỗi bit P_E**
BER dựa vào việc thực hiện K lần thí nghiệm, mỗi lần truyền qua kênh N ký hiệu, mỗi lần đếm được N_e lỗi trong số N ký hiệu được phát, giá trị của N_e tại mỗi lần trong số K lần thực hiện thí nghiệm thường khác nhau (do tính ngẫu nhiên của kênh gây ra). Hay nói cách khác, tái tạo lại thí nghiệm ngẫu nhiên bằng cách phát N ký hiệu qua kênh tạp âm (ngẫu nhiên) K lần, thường nhận được K kết quả đếm lỗi N_e khác nhau.	Xác suất lỗi bit là dựa trên việc truyền vô hạn các ký hiệu qua hệ thống $P_E = \lim_{N \to \infty} \frac{N_e}{N}$
Phát hữu hạn các ký hiệu qua hệ thống (N hữu hạn)	Phát vô hạn các ký hiệu qua hệ thống ($N \to \infty$)
⇒ Tỉ số lỗi bit BER là một biến ngẫu nhiên (ngẫu nhiên là do giá trị N_e tại mỗi lần đếm lỗi khác nhau). Tỉ số lỗi bit: $BER = \hat{P}_E = \frac{N_e}{N} = \frac{\text{Số ký hiệu bị lỗi}}{\text{Số ký hiệu được phát}}$ là một tỉ số.	⇒ Xác suất lỗi bit P_E là một con số (tất định), ví dụ hệ thống BPSK $P_E = Q\left(\sqrt{2E_b/N_0}\right)$. Tất cả đều là tất định.
BER là một ước tính của xác suất lỗi bit	

Vì các thuật ngữ *tỉ số lỗi bit* BER và *xác suất lỗi bit* thường lấy trung bình cùng một sự kiện nên dễ gây nhầm lẫn giữa chúng. Tuy nhiên, thực tế chúng khá khác nhau bởi lẽ: *(i) Tỉ số lỗi bit* BER là một ước tính của xác suất lỗi bit. Nên nhớ rằng "*tỷ số*" được tạo ra là một phân số (ví dụ số km/giờ). BER là một *tỷ số* (*N_e lỗi/N ký hiệu phát*). Tái tạo lại thí nghiệm ngẫu nhiên bằng cách phát N ký hiệu qua kênh tạp âm (ngẫu nhiên) K lần, thường nhận được K kết quả đếm lỗi N_e khác nhau; *(ii) Xác suất lỗi bit* là dựa trên việc truyền *vô hạn* các ký hiệu qua hệ thống. Xác suất lỗi bit là một con số chứ không phải là biến ngẫu nhiên. Ví dụ xác suất lỗi bit đối với hệ thống BPSK trong kênh AWGN là $P_E = Q\left(\sqrt{2E_b/N_0}\right)$, trong đó E_b là năng lượng

trên bit và N_0 là mật độ phổ công suất một phía của tạp âm kênh. Con số này giữ cố định miễn là E_b và N_0 được giữ không đổi. Sự khác nhau cơ bản giữa tỉ số lỗi bit BER và xác suất lỗi bit P_E được cho ở bảng 1.1

Giả sử ta thực hiện K = 7 lần mô phỏng Monte Carlo độc lập của một hệ thống truyền thông BPSK trong đó ta đã điều chỉnh E_b/N_0 sao cho xác suất lỗi ký hiệu là 0,1. Mỗi mô phỏng dựa vào phát đi $N = 1000$ ký hiệu. Kết quả của việc tái tạo lại thí nghiệm ngẫu nhiên về việc cho 1000 ký hiệu qua kênh ngẫu nhiên 7 lần được cho trong hình 1.9. Tính ngẫu nhiên là hiển nhiên mà BER dựa trên bất kỳ số truyền dẫn $N \leq 1000$ tạo ra dải các kết quả. Tổng quát, hiện tượng trải rộng kết quả này liên quan với phương sai của ước tính và để các kết quả mô phỏng là hữu hiệu thì sự trải rộng nên là nhỏ. Lưu ý rằng các kết quả được cho ở hình 1.9 có phương sai nhỏ hơn khi N lớn hơn. Đây là tính cách điển hình đối với bộ ước tính được triển khai chính xác. Cũng cần lưu ý rằng, khi N lớn thì các kết quả co về xác suất đúng và ta có xu hướng tin rằng, khi mô phỏng được triển khai đúng, thì bộ ước tính $\hat{P}_E$ sẽ hội tụ về xác suất lỗi P_E phù hợp với định nghĩa tần suất tương đối của xác suất. Đây cũng là đặc trưng của các bộ ước tính được triển khai chính xác. Hai điều kiện mong muốn này là các khái niệm được định rõ trong lý thuyết ước tính. Nếu phương sai của ước tính dẫn tới 0 khi N lớn tuỳ ý thì ta nói rằng ước tính là ***kiên định***. Cũng vậy, nếu $E\{\hat{P}_E\} = P_E$ thì ta nói rằng ước tính là ***không chệch***. Ta sẽ đề cập nhiều về các tính chất của các bộ ước tính ở các chương sau, và ta cũng sẽ học cách để triển khai mô phỏng dựa trên hình 1.9.



Hình 1.9: Các kết quả mô phỏng Monte Carlo

1.6. Vai trò của mô phỏng

Mô phỏng được sử dụng rộng rãi trong nhiều giai đoạn của quá trình thiết kế và triển khai hệ thống truyền thông hiện đại. Mô phỏng *không những* chủ yếu được sử dụng để *ước lượng* hiệu năng và nghiên cứu dung hoà trong thiết kế (tối ưu tham số) *mà còn* được dùng để *thiết lập* các thủ tục kiểm tra và kiểm chuẩn, các dự đoán tuổi thọ cũng như nghiên cứu tính bất thường sau khi hệ thống được triển khai trong thực tế. Cả phương pháp luận mô phỏng và mô hình mô phỏng đều được sử dụng để biểu diễn hệ thống phụ thuộc vào các giai đoạn khác nhau của quá trình thiết kế, thực thi và vòng đời của hệ thống. Phương pháp luận mô phỏng cũng sẽ được kiểm soát hoặc định hướng bởi trình tự thiết kế tổng thể. Ta minh họa trình tự thiết kế và sử dụng mô phỏng trong các giai đoạn thiết kế khác nhau và vòng đời của hệ thống truyền thông.

Quá trình thiết kế một hệ thống truyền thông phức tạp được thực hiện từ "*đỉnh xuống*", ngược lại thực thi phần cứng thường từ đáy lên. Theo đó, khi thiết kế hệ thống ta bắt đầu tại mức hệ thống (mức trừu tượng cao nhất) và hoàn thiện chi tiết thiết kế từ mức hệ thống, xuống mức hệ thống con và cuối cùng là mức thành phần. Sau đó đạt tới mức đáy mà ở đó tính chi tiết của bộ phận thiết bị thành phần được nhận dạng. Vì vậy, trình tự xây dựng một hệ thống gồm: *(i)* Tạo ra các thành phần; *(ii)* Hợp các thành phần thành hệ thống con; *(iii)* Xây dựng hệ thống thổng thể từ các hệ thống con; *(iv)* Triển khai mô phỏng theo giải pháp từ đỉnh xuống. Ta bắt đầu bằng việc mô phỏng mức hệ thống có mức trừu tượng cao tăng dần tính chi tiết, các mô phỏng các hệ thống con và các thành phần. Do thực thi khởi đầu nên các đặc tính đo của các thành phần và các phân hệ được chứa trong mô hình mô phỏng.

Tại đây ta mô tả các giai đoạn khác nhau của quá trình thiết kế và cách sử dụng các mô phỏng trong các giai đoạn khác nhau của quá trình thiết kế.

1.6.1. Quá trình đặc tả kỹ thuật mức hệ thống và độ dự trữ tuyến

Quá trình thiết kế một hệ thống truyền thông bắt đầu bằng việc trình bày, phân tích các yêu cầu người dùng và hiệu năng mong muốn gồm: thông lượng, tỷ số lỗi, xác suất ngừng hoạt động, hạn chế băng thông, công suất, trọng lượng, phức tạp/chi phí, kênh trên đó hệ thống hoạt động, và tuổi thọ trung bình của hệ thống. Trên cơ sở những yêu cầu của người dùng, "*kỹ sư hệ thống*" xuất phát từ khái niệm ban đầu về hệ thống như: sơ đồ điều chế, kỹ thuật mã hoá, cân bằng nếu cần... Tập giá trị tham số gọi là các *chỉ tiêu kỹ thuật mức A* như các mức công suất, băng thông, chỉ số điều chế cũng được thiết lập trong giai đoạn khởi đầu của quá trình thiết kế.

Mục đích toàn diện tại điểm này là: *(i)* xác định cấu hình hệ thống; *(ii)* các giá trị của tham số (sẽ đáp ứng các mục đích hiệu năng và thoả mãn những ràng buộc thiết kế). Thấy rõ, hiệu năng hệ thống là một hàm của SNR (tương đương E_b/N_0) và méo tổng do tất cả các thành phần trong tuyến truyền thông. SNR được thiết lập thông qua một quá trình gọi là ***dự trữ tuyến***, là phần tính toán mạnh nhất đưa vào bản mô tả các nhân tố như: công suất phát, hệ số khuếch đại ăng ten, suy hao đường truyền, khuếch đại công suất, hệ số tạp âm của bộ khuếch đại và bộ lọc. Trong khi dự trữ tuyến không phải là đại lượng chính trong mô phỏng, nhưng nó thiết lập một dải các giá trị SNR hoặc E_b/N_0 trên đó phải được mô phỏng để ước tính hiệu năng.

Vì không thể xây dựng các thành phần lý tưởng nên việc thực thi các thành phần thực tế như bộ khuếch đại và bộ lọc sẽ tạo ra tính không lý tưởng. Dẫn đến méo tín hiệu, sẽ ảnh hưởng

lên hiệu năng hệ thống. Chúng phải được tính đến trong dự dữ tuyến bằng cách tính toán hiệu năng của hệ thống theo các thành phần lý tưởng, sau đó cộng với "*tổn thất thực thi*". Tổn thất thực thi đánh giá sự tăng E_b/N_0 để vượt qua ảnh hưởng của méo do các thành phần không lý tưởng gây ra. Đôi khi tổn thất thực thi được xem là các tham số truyền thông hoặc tham số méo. Lưu ý rằng một số tham số như băng thông bộ lọc ảnh hưởng lên công suất tạp âm tại các điểm khác nhau trong hệ thống và sẽ lần lượt ảnh hưởng lên độ dự trữ và méo.

Người thiết kế hệ thống bắt đầu bằng cấu hình khởi đầu cho hệ thống, các chỉ tiêu kỹ thuật mức A và độ dự trữ tuyến. Dự trữ tuyến thường được biểu diễn ở dạng bảng tính và dòng cuối cùng trong dự trữ tuyến là E_b/N_0 mạng tại điểm tới hạn trong hệ thống sau khi tất cả các tổn thất thực thi được tính đến. "*Điểm tới hạn*" này thường là đầu vào máy thu. Dự trữ tuyến được gọi là "*kín*" hay "*cân bằng*" nếu tuyến có E_b/N_0 đủ lớn với hệ số dự phòng an toàn để tạo hiệu năng hệ thống chấp nhận được. Tồn tại nhiều phép đo hiệu năng hệ thống chẳng hạn: *(i)* ở hệ thống tương tự thường dùng lỗi trung bình bình phương MSE; *(ii)* ở hệ thống số, phép đo hiệu năng điển hình là tỷ số lỗi bit BER. Tại điểm này, số đo hiệu năng được tính toán từ các công thức xấp xỉ và chưa được mô phỏng. Vì tất cả những tổn thất thực thi đã được tính đến trong E_b/N_0 mạng nên có thể tính BER theo công thức hệ thống lý tưởng.

Nếu độ dự trữ tuyến *không kín* hay *không cân bằng* thì các chỉ tiêu kỹ thuật mức A, các tổn thất thực thi và thậm chí cấu hình hệ thống bị thay đổi và độ dự trữ tuyến được tính toán lại. Ví dụ băng thông của bộ lọc có thể thay đổi, kích thước ăng ten có thể tăng và đặc tính kỹ thuật hệ số tạp âm của bộ khuếch đại có thể thấp hơn. Quá trình này xảy ra liên tục cho đến khi độ dự trữ tuyến được cân bằng với hệ số dự trữ thích hợp.

Trên cơ sở cấu hình hệ thống khởi đầu, các đặc tính kỹ thuật mức A và độ dự trữ tuyến, khi này được coi là kín, cho phép xây dựng mô hình mô phỏng để kiểm tra độ dự trữ tuyến và *tinh chỉnh* thiết kế. Có thể ước tính chính xác các phép đo hiệu năng và kiểm tra suy thoái hiệu năng do thực thi không lý tưởng thông qua các mô phỏng chi tiết. Nếu những phân bổ trong độ dự trữ tuyến được kiểm tra thông qua mô phỏng và độ dự trữ tuyến vẫn được kín thì quá trình thiết kế chuyển sang giai đoạn kế tiếp mà ở đó chứa thiết kế chi tiết, thực thi các hệ thống con và các thành phần. Nếu độ dự trữ tuyến *không kín* thì một số phân bổ méo, cấu hình hệ thống và đặc tính kỹ thuật mức A có thể bị thay đổi.

Giai đoạn đầu của thiết kế liên quan "*nghệ thuật*" và thuộc về người có kinh nghiệm trong thiết kế hệ thống truyền thông. Hầu hết các trường hợp thiết kế ban đầu dựa vào những thiết kế trước với những sửa đổi nhỏ.

1.6.2. Thực thi và kiểm tra các thành phần then chốt

Thiết kế một hệ thống truyền thông mới sẽ luôn chứa một số thuật toán xử lý tín hiệu mới và các công nghệ phần cứng (phần mềm) mới. Bất kỳ công nghệ mới luôn có rủi ro hoặc không chắc chắn về hiệu năng. Nếu công nghệ mới có một phần tử tới hạn thì trước hết thành phần đó phải được xây dựng và kiểm tra trong các điều kiện thực tế nhằm kiểm tra hiệu năng và giảm thiểu rủi ro. Vì chỉ vài thành phần then chốt được xây dựng tại giai đoạn đầu của tiến trình thiết kế nên không thể kiểm tra toàn bộ hệ thống phần cứng. Trong tình huống này, mô phỏng tạo ra

môi trường kiểm tra tuyệt vời và sử dụng mô phỏng ít tốn kém hơn nhiều so với dùng phần cứng để *nguyên mẫu hóa* toàn bộ hệ thống. Tất cả các thành phần và tín hiệu được mô phỏng với các đặc tính đo phải được đưa vào mô hình mô phỏng. Ví dụ, nếu thành phần sẽ được kiểm tra là bộ khuếch đại mới, thì các đặc tính hàm truyền đạt AM-AM và AM-PM của nó được đo và các đặc tính này được đưa vào mô hình phi tuyến cho bộ khuếch đại đó. Sau đó mô phỏng toàn bộ hệ thống để kiểm tra hiệu năng và độ dự trữ tuyến. Một lần nữa nếu đưa các đặc tính đo vào mô phỏng cho thấy méo tốt hơn mong đợi thì lưu lại ở đâu đó trong hệ thống.

Nếu độ dự trữ tuyến kín thì việc triển khai phần cứng đi đến thành phần tới hạn kế tiếp. Nếu không thì thiết kế lại thành phần, xây dựng lại và kiểm tra lại hoặc độ dự trữ tuyến được sửa đổi để tính đến suy thoái phụ do thành phần (ngoài những gì đã được phân bổ trong độ dự trữ tuyến cho thành phần đó). Quy trình này được lặp lại đối với các thành phần then chốt khác.

1.6.3. Hoàn thiện nguyên mẫu phần cứng và phê chuẩn mô hình mô phỏng

Như thủ tục đã được mô tả, nguyên mẫu phần cứng của toàn bộ hệ thống bắt đầu xuất hiện cùng với mô hình mô phỏng kèm theo. Tại đây, mô hình mô phỏng bao hàm các đặc tính đo cho hầu hết các thành phần trong hệ thống. Nhiều số đo hiệu năng cho toàn hệ thống được thực hiện trên nguyên mẫu phần cứng này. Các mô phỏng song song cũng được kiểm soát. Có thể so sánh các đặc tính hiệu năng đo với các kết quả mô phỏng và ngược lại. Mô phỏng tạo ra những điểm chuẩn để kiểm tra và kết quả kiểm tra sẽ phê chuẩn mô phỏng. Kết quả cuối cùng của quá trình thiết kế này là nguyên mẫu hoàn chỉnh của hệ thống, cho ta cơ sở để triển khai phiên bản sản phẩm của hệ thống. Ngoài ra, một khi có mô hình mô phỏng được phê chuẩn, có thể dự đoán tuổi thọ với mức tin tưởng cao.

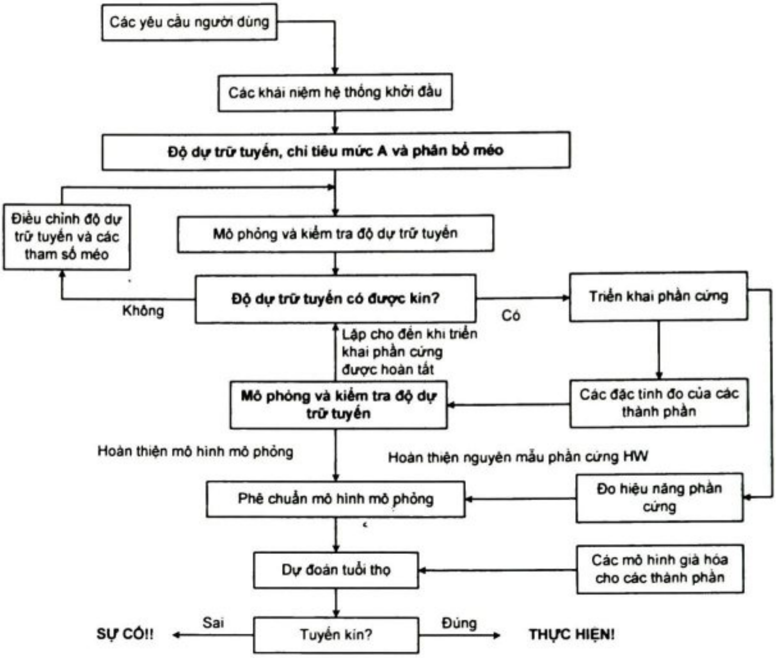
1.6.4. Dự đoán tuổi thọ

Trong khi quy trình trên dẫn đến thiết kế đảm bảo mức hiệu năng cho trước khi triển khai hệ thống thì một yêu cầu quan trọng nữa phải được đáp ứng cho hầu hết các hệ thống, là hiệu năng tuổi thọ. Nhiều hệ thống truyền thông như vệ tinh truyền thông, hệ thống cáp biển được mong đợi phải có tuổi thọ dài (thường khoảng 10 năm hoặc hơn) trong khoảng thời gian này hiệu năng phải được đảm bảo. Tất nhiên, không thể kiểm tra vòng đời thực tế dựa trên nguyên mẫu phần cứng. Trong khi các thủ tục kiểm tra vòng đời tăng tốc đã được triển khai, thực tế thường dùng mô phỏng như giải pháp bổ sung để kiểm tra vòng đời tăng tốc.

Các dự đoán hiệu năng tuổi thọ dùng mô phỏng bằng các mô hình già hóa cho các thành phần chính của hệ thống. Nếu ta có mô hình mô phỏng đã được phê chuẩn cho toàn bộ hệ thống tại lúc bắt đầu vòng đời và cũng có những mô hình tính cách tốt của các thành phần như hàm tuổi thì các mô hình già hóa có thể được thay thế trong mô hình bắt đầu vòng đời để đạt được các số đo hiệu năng tuổi thọ cho hệ thống.

Nếu hiệu năng dự đoán tuổi thọ là thoả đáng và dự trữ tuyến cuối đời là kín với hệ số dự phòng thoả đáng thì việc thiết kế và thực thi hệ thống là hoàn thiện. Ngược lại quá trình phải lặp lại cho đến khi đạt được hội tụ.

Tóm tắt các bước then chốt trong trình tự thiết kế và vai trò của mô phỏng trong thiết kế các hệ thống truyền thông được minh họa ở hình 1.10.



Hình 1.10: Trình tự thiết kế và vai trò mô phỏng

1.7. Gói phần mềm để mô phỏng và khuyến nghị

Gói phần mềm để mô phỏng

Trong những thập kỷ qua nhiều gói phần mềm đã được triển khai và đang được sử dụng rộng rãi để mô phỏng các hệ thống truyền thông ở mức dạng sóng. Các thành phần cốt yếu của cấu trúc khung mô phỏng cho các hệ thống truyền thông gồm: *(i)* bộ xây dựng mô hình; *(ii)* thư viện mô hình; *(iii)* nhân mô phỏng; *(iv)* bộ hậu xử lý.

Bất kể dùng gói mô phỏng nào đi nữa, bước đầu tiên trong mô phỏng hệ thống truyền thông gồm: *(i)* xây dựng các mô hình mô phỏng cho các phân hệ khác nhau; *(ii)* cấu hình hóa các phân hệ thành một mô phỏng hệ thống đầu cuối - đầu cuối. Các mô hình mô phỏng có thể được xây dựng bằng ngôn ngữ lập trình mục đích chung và viết mã thích hợp hoặc sử dụng bộ xây dựng mô hình đồ họa. Với bộ xây dựng đồ họa, triển khai các mô hình mô phỏng cho các phân hệ và cho toàn hệ thống truyền thông các khối cơ bản có sẵn từ các thư viện mô hình. Các biểu tượng biểu diễn các khối chức năng như các nguồn tin, bộ mã hoá, bộ điều chế, bộ ghép/phân kênh, các mô hình kênh, tạp âm và nguồn nhiễu, bộ lọc, bộ giải điều chế, bộ giải mã được chọn từ các thư viện mô hình. Kết nối các biểu tượng này sẽ tạo mô hình mô phỏng ở

dạng sơ đồ khối phân cấp. Điển hình như Simulink trong Matlab là gói mô phỏng tương đối đơn giản sử dụng giải pháp bộ xây dựng mô hình đồ họa.

Các mô hình được xây dựng hoặc từ đỉnh xuống hoặc từ đáy lên trong đó xây dựng từ đỉnh xuống là thích hợp cho thiết kế hệ thống và giải pháp từ đáy lên là sự lựa chọn của thiết kế phần cứng. Tại *"mức lá"* là mức nhỏ nhất trong phân cấp, các mô hình có thể có một số biểu diễn trong phạm vi từ các thường trình con dấu phẩy động hoặc các thủ tục trong ngôn ngữ lập trình như FORTRAN, C, C++ tới thực hiện mức bit của các mô hình hệ thống con trong VHDL.

Giải pháp thay thế dùng trình soạn thảo sơ đồ khối đồ họa để xây dựng mô hình là dùng ngôn ngữ trung gian như ngôn ngữ lệnh Matlab. Việc tạo những mô phỏng để định hướng triển khai các hệ thống truyền thông phức tạp và tốn kém phù hợp với giải pháp sơ đồ khối và các bộ xây dựng mô hình đồ họa. Bởi lẽ giải pháp sơ đồ khối là một trình bày mang tính bản chất của các hệ thống truyền thông và tạo môi trường sử dụng thân thiện cho kỹ sư hệ thống.

Mở rộng nỗ lực xây dựng mô hình mô phỏng được quy vào tính khả dụng của các thư viện mô hình. Nhiều gói mô phỏng thương mại cho các hệ thống truyền thông khả dụng ngày nay có các thư viện mô hình mở rộng khả dụng.

Sau khi mô hình mô phỏng được triển khai thì các *tham số mô phỏng* (như: tốc độ lấy mẫu, các con số gốc của bộ tạo số ngẫu nhiên và thời gian mô phỏng) và các *tham số thiết kế* (như: độ rộng băng tần bộ lọc, tỉ lệ mã, SNR) phải được định rõ. Sau đó thực hiện mô phỏng. Liên kết tất cả các mô hình với nhau sẽ tạo mã khả thi (có khả năng thực hiện mô phỏng), bắt đầu mô phỏng, lưu các giá trị mẫu của các dạng sóng được tạo bởi mô phỏng và giám sát tiến trình mô phỏng là các hàm thường được thực hiện bởi nhân mô phỏng/bộ quản lý.

Sau khi hoàn tất mô phỏng, tính các phép đo hiệu năng như BER, SNR từ các dạng sóng mô phỏng, sử dụng các *"bộ hậu xử lý"* để trực quan hóa kết quả mô phỏng là hàm của các tham số thiết kế. Cho phép khảo sát phổ, dạng sóng, biểu đồ tán xạ, biểu đồ mắt theo các tham số đặc trưng của mô hình mô phỏng một cách chi tiết, trực quan và gỡ rối mô phỏng.

Nhờ khả năng đồ họa tương tác hỗ trợ đắc lực cho phân tích, thiết kế, nâng cấp, mở rộng mô hình mô phỏng cũng như so sánh đánh giá hiệu năng giữa chúng.

Các cốt lõi mô phỏng tạo ra các kỹ thuật lấy mẫu và mô phỏng khác nhau, chúng thường được phân loại là điều khiển thời gian (lấy mẫu một tốc độ, lấy mẫu đa tốc độ hoặc lấy mẫu tốc độ khả biến), điều khiển luồng, điều khiển sự kiện hoặc kết hợp.

Trường hợp đơn giản nhất là mô phỏng điều khiển thời gian, khi này có một đồng hồ mô phỏng và mỗi khối chức năng trong mô hình mô phỏng được thực hiện tại mỗi *"tick"* của đồng hồ mô phỏng. Sau đó đồng hồ mô phỏng được nhịp lên một chu kỳ lấy mẫu (chu kỳ đồng hồ bằng chu kỳ lấy mẫu). Tất cả các khối chức năng trong mô hình được kích hoạt tại mỗi chu kỳ đồng hồ vì vậy mỗi mô hình có thể cập nhật trạng thái ứng với giá trị mới của đồng hồ mô phỏng. Mô phỏng loại này được xây dựng ở dạng vòng *"do"* hoặc vòng *"for"* trong đó tại mỗi nhịp của đồng hồ mô phỏng chỉ số vòng lặp được tăng lên một đơn vị.

Trường hợp mô phỏng điều khiển sự kiện, đặt trước đồng hồ một lượng nào đó cho thời gian biểu của sự kiện tiếp theo và mỗi khối chức năng cập nhật trạng thái ứng với giá trị của thời điểm mô phỏng mới. Một cách điển hình, chỉ vài khối cần được kích hoạt để cập nhật trạng riêng và không cần xử lý trong khoảng thời gian *"liên kết sự kiện"*. Mô phỏng các hệ thống hàng đợi thường triển khai theo cách này.

Các mô phỏng điều khiển sự kiện và mô phỏng kích thước bước khả biến có hiệu quả tính toán cao hơn mô phỏng điều khiển thời gian. Tuy nhiên, trong một số trường hợp cần phải nội suy và lấy mẫu lại và chúng chứa thông tin mào đầu liên quan tới lập lịch. Mô phỏng các hệ thống truyền thông thường dùng nhất là mô phỏng điều khiển thời gian với lấy mẫu đơn tốc hoặc đa tốc độ. Lấy mẫu đa tốc để mô phỏng tín hiệu hệ thống có băng thông thay đổi rộng, điển hình là hệ thống trải phổ, ở đó ta sử dụng lấy mẫu đa tốc độ nhằm giảm thiểu thời gian mô phỏng.

Các thuật toán xử lý tín hiệu số đóng vai trò quan trọng trong mô phỏng và thực thi hệ thống truyền thông. Vì vậy, cần quan tâm độ rộng bit, chia sẻ tài nguyên trong mô hình mô phỏng và cứng hóa mô hình mô phỏng. Khi thực thi phần cứng bằng cách sử dụng các ngôn ngữ mô tả phần cứng như VHDL để tạo ra giao diện giữa cơ cấu mô phỏng mức hệ thống và các công cụ thiết kế phần cứng. Khi mềm hóa phân hệ, cơ cấu mô phỏng có thể biên dịch thuật toán mô phỏng vào mã ngôn ngữ Assembly cho bộ xử lý DSP đích. Những liên kết để thực thi này đang ngày càng quan trọng hơn bao giờ hết khi mà càng nhiều hàm trong các máy thu truyền thông được thực hiện trong phần cứng số hoặc như phần mềm nhúng.

Khuyến nghị

Ta không nên nghĩ mô phỏng là sự thay thế phân tích hoặc các phép đo phần cứng truyền thống. Mô phỏng chỉ hữu hiệu nhất khi được sử dụng với mục đích phân tích và đo. Bằng cách thực hiện mô phỏng nhiều lần cho phép phát hiện các tham số tới hạn và đơn giản hóa hệ thống. Khi đơn giản hoá cần phải có phân tích bổ sung.

Cần có mức độ phân tích nhất định để giải các bài toán mức hệ thống. Ví dụ: phải hiểu rõ tính phụ thuộc cơ bản của các tham số hiệu năng như BER, lỗi trung bình quân phương MSE tại đầu ra bộ giải điều chế hoặc SNR tại đầu vào máy thu; các tham số hệ thống như công suất phát và băng thông, dạng điều chế hoặc tỷ lệ mã để đảm bảo hệ thống đang hoạt động đúng và các kết quả mô phỏng là hợp lý. Nói cách khác khi các tham số thay đổi trong mô phỏng, ta phải đảm bảo rằng các kết quả quan sát về những thay đổi này là hợp lý và kiên định theo lý thuyết đã biết. Những *"kiểm tra khôn khéo"* là quan trọng để phê chuẩn mô phỏng và luôn cần có các nỗ lực phân tích nhất định.

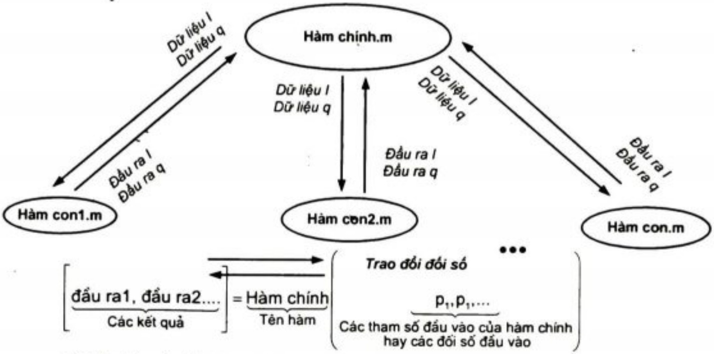
1.8. Sử dụng Matlab

Matlab sẽ được sử dụng để giải thích các khái niệm, giải quyết các vấn đề và thực hiện các mô phỏng mẫu. Một số lý do cho sự lựa chọn Matlab:

Thứ nhất, Matlab là ngôn ngữ lập trình bậc cao, ngôn ngữ kỹ thuật, ngôn ngữ chuyên gia có thư viện toán cực mạnh, giao diện đồ họa phong phú, khả năng thích ứng các ngôn ngữ khác, cho phép tạo giao diện người dùng tiện lợi. Đặc biệt cho phép ứng dụng kết nối điều khiển thiết bị phần cứng, thực tế rất nhiều các hãng sản suất thiết bị viễn thông như: hãng sản xuất thiết bị đo Agilent Technology (được tách ra từ HP), hãng Motorola... ở đó việc thiết kế, sản suất thiết bị hệ thống trên cơ sở phân lớp công đoạn sản suất chẳng hạn: Xét hệ thống theo quan điểm chức năng, sau khi công thức hoá các khối chức năng, xây dựng quan hệ hàm biến cho các khối chức năng, đánh giá khối chức năng theo các tiêu chí đặc thù (các thông số tối ưu hoá, các tiêu chuẩn đánh giá), mô phỏng khối chức năng trong nhiều miền tín hiệu (miền tần số, miền thời gian, miền Z...) thông qua các biểu đồ, đồ thị... Tiếp theo là công đoạn thiết kế các mạch điện cụ thể cho các khối chức năng, trong công đoạn này tính liên kết của Matlab được thể hiện cực

mạnh ở đây, Matlab cho phép liên kết với các chương trình thiết kế mạch điện chuyên dụng như ÇiruitMaker.... Cuối cùng sản xuất thiết bị viễn thông trong dây chuyền sản xuất, các thiết bị được đánh giá thông qua tín hiệu phản hồi trở về Matlab cứ như thế cho đến khi đạt được chất lượng mong muốn. Matlab có ưu việt nổi trội là khả năng mô phỏng hệ thống động, cho phép thể hiện tín hiệu và hệ thống trong nhiều miền xét (miền thời gian, miền tần số, miền Z...). Hơn nữa, tính cô đọng (nhiều thuật toán phức tạp được trình bày bằng rất ít dòng lệnh của mã), sự hỗ trợ của đồ họa, tính modul...

Thứ hai, Matlab được dùng rộng rãi trong các chương trình đào tạo kỹ sư, hầu hết các sinh viên thực sự đã có các tài nguyên cần thiết để thực hiện chương trình Matlab. Với những chương trình Matlab có tính dung hòa giữa khối lượng tính toán lớn và thời gian cần thiết để chạy chương trình phù hợp, thì nó cần phải dùng một ngôn ngữ biên dịch như C hoặc C++. Đặc biệt có ý nghĩa khi dùng các mô phỏng Monte Carlo để ước tính BER tại SNR lớn, vì khi đó phải xử lý nhiều ký hiệu qua kênh để đạt được bộ ước tính BER chất lượng (ước tính không chệch và kiên định). Tuy nhiên, Matlab là một công cụ cực mạnh trong tình huống này, vì mô phỏng nguyên mẫu được triển khai trong Matlab để thiết kế và kiểm chứng các giải thuật xử lý tín hiệu DSP riêng cũng như mô phỏng toàn diện. Mã chương trình Matlab kết quả có thể ánh xạ vào mã chương trình C hoặc C++ nhằm thực hiện hiệu quả hơn và kết quả đạt được so sánh với các kết quả thực hiện bằng Matlab. Dùng Matab để nguyên mẫu hóa cho phép nhận biết các lỗi khái niệm một cách nhanh chóng, làm tăng tốc phát triển phần mềm cuối cùng.



% Bắt đầu của file hàm chính

Biến đầu ra 1 = phương trình tính toán dùng các đối số đầu vào ⇔ Hàm con

Biến đầu ra 2 = phương trình tính toán dùng các đối số đầu vào ⇔ Hàm con

:

% kết thúc file hàm chính

Hình 1.11: Minh họa tính phân lớp hàm - Quan hệ giữa các file hàm chính và hàm con

Thứ ba, với mục đích làm sáng tỏ các giải thuật được dùng trong quá trình mô hình hóa và mô phỏng hệ thống cũng như việc đánh giá kết quả mô phỏng, vì vậy sẽ tập trung khai thác

thư viện toán của Matlab, hay nói cách khác sử dụng các hàm cơ bản có trong thư viện, khai thác tối đa khả năng phân lớp hàm của Matlab để viết chương trình mô phỏng. Theo đó, các chương trình mô phỏng được thực hiện trên thuật toán sử dụng ngôn ngữ lập trình Matlab, chúng được viết dưới dạng các file.m. Trong một chừng mực nhất định, điều này được minh họa ở hình 1.11. Khả năng phân lớp này rất phù hợp với tính phân lớp của mô hình mô phỏng. Lưu ý rằng ta có thể dùng biến toàn cục. SIMULINK mặc dù được thiết kế cho mô phỏng, nhưng không được đề cập ở đây vì nó không phù hợp để trình bày tính chi tiết của các thuật toán trong các chương trình mô phỏng cũng như phương pháp luận để triển khai mã chương trình.

Cuối cùng là, Matlab được sử dụng rộng rãi trong cộng đồng thiết kế kỹ thuật. Matlab kết hợp khả năng tính toán tuyệt vời với khả năng đồ họa thân thiện. Matlab chứa đựng một thư viện phong phú các hàm tiền lập trình (các file.m) để: tạo, phân tích, xử lý và hiển thị tín hiệu. Các thư viện bổ sung (các hộp công cụ) cho phép bổ sung vào thư viện Matlab cơ bản các file.m để định rõ các lĩnh vực ứng dụng. Cho phép người dùng Matlab dễ dàng tạo ra file.m mới phục vụ ứng dụng riêng. Ngoài ra mã chương trình Matlab rất súc tích, ngắn gọn, làm cho nó có khả năng biểu diễn các thuật toán xử lý tín hiệu phức tạp chỉ vài dòng mã.

Có lẽ vì thế mà hầu hết các trường Đại học, các Viện nghiên cứu lớn đều chọn Matlab để phục công tác đào tạo, nghiên cứu phát triển, đặc biệt thực thi hoá kết quả công trình nghiên cứu vào thực tế.

Vì mục đích trình bày phương pháp luận mô phỏng, thực thi mô phỏng trong thiết kế một cách cơ bản và dễ hiểu nhất có thể cũng như giới hạn khôn khổ cuốn sách, nên các hệ thống được xét để làm ví dụ minh họa mới chỉ dừng ở mức đơn giản ở dạng riêng biệt đơn giản trong nhiều giả định lý tưởng. Tất cả đều thuộc loại không thích ứng, vì vậy từ góc độ sử dụng tài nguyên là không hiệu quả. Như một minh họa đơn giản nhất: do tính cách thay đổi ngẫu nhiên theo thời gian của môi trường truyền làm tín hiệu thu bị thăng giáng ngẫu nhiên, vì thế trong thiết kế để đảm bảo chất lượng BER ta thường phải đưa thêm độ dự trữ pha đinh hoặc dùng sơ đồ điều chế BPSK (hiệu năng thông lượng thấp)... cho trường hợp kịch bản kênh tồi, nhưng tồn tại rất nhiều thời điểm kịch bản kênh tốt, khi này có thể dùng các sơ đồ điều chế 4-QAM, 16-QAM,... nhưng vẫn đảm bảo được chất lượng BER, vì vậy ta đã không khai thác triệt để tài nguyên dung lượng hệ thống. Nhiều chương trình mô phỏng được viết trên Matlab, được trình bày trong phần văn bản, việc quyết định đưa mã chương trình Matlab vào trong nội dung cuốn sách với mục đích: *(i)* làm sáng tỏ việc sử dụng phương pháp luận để triển khai mô phỏng cũng như minh họa các thuật toán xử lý tín hiệu số DSP quan trọng; *(ii)* nhiều đoạn mã chương trình Matlab trong các ví dụ sẽ hỗ trợ mở rộng triển khai mô hình và mô phỏng riêng của bạn đọc. Tuy nhiên, để không làm gián đoạn tính liên tục của cuốn sách, chỉ có những chương trình ngắn (dài không quá một trang trang văn bản) thực hiện: định nghĩa các thông số đặc trưng, liên kết các chương trình con (ở dạng file.m), các đặc trưng vào ra của, các kết quả mô phỏng được trình bày. Với những chương trình quá dài, để phù hợp với một trang văn bản chương trình được liên kết phù hợp và được lưu trên đĩa CD-ROM.

Chương 2

PHƯƠNG PHÁP LUẬN MÔ PHỎNG

2.1. Mở đầu

Mô phỏng đóng vai trò quan trọng trong thiết kế hệ thống truyền thông, được dùng để thiết kế chi tiết các phần tử và ước lượng hiệu năng mức hệ thống. Chương này sẽ trình bày phương pháp luận cơ bản để triển khai mô phỏng, và chi tiết hóa quá trình lập mô hình và mô phỏng cũng như vấn đề định lượng và định tính của mô phỏng. Một cách trình bày khác, mô phỏng vừa có *tính nghệ thuật* vừa có *tính khoa học*.

Tồn tại một số bước tạo và thực thi mô hình mô phỏng trên cơ sở lý thuyết, vì vậy có bản chất định lượng chẳng hạn: mô hình hóa các phần tử hệ thống riêng biệt và tạo các số ngẫu nhiên. Mặt khác, nhiều bước trong mô phỏng bao gồm các giải pháp và các xem xét mà không thể xác định số lượng rõ ràng, là bản chất khám phá. Chúng được gộp lại một cách khá lỏng lẻo, được coi là ***"Phương pháp luận"*** mô phỏng. Điểm nhấn của chương này là phương pháp luận hay ***"Tính nghệ thuật"*** của mô phỏng, đặc biệt ý nghĩa khi ước lượng hiệu năng ở mức hệ thống. ***"Tính khoa học"*** của mô phỏng bàn về khía cạnh định lượng của việc mô hình hóa, ước tính,.. sẽ được giải quyết trong các chương sau. Các phần định tính và định lượng không hẳn tách rời nhau mà quan hệ khá mật thiết với nhau. Tất cả các bước trong mô phỏng bao gồm mô hình hóa các thành phần cụ thể và một số *"phương pháp luận"*. Hơn nữa việc thực hiện mô phỏng cần đến một tập các thuật toán.

Với mục đích trình bày và thảo luận: Chương này sẽ đề cập 2 chủ đề, các mặt *định tính* và *định lượng* của mô phỏng như thể chúng tách rời nhau. Tuy nhiên, việc nghiên cứu chúng có tính hỗ trợ nhau. Vì vậy, ta nên đọc chương này trước khi tiếp các chương sau. Chương này có tính logic và lý luận cao.

Bài toán mô phỏng đơn giản nhất thường gồm 4 bước cơ bản sau:

- Ánh xạ bài toán đã cho thành mô hình mô phỏng.
- Phân giải bài toán tổng thể thành một tập các bài toán nhỏ hơn.
- Chọn tập các kỹ thuật mô hình hóa, mô phỏng, ước tính phù hợp và áp dụng chúng để giải quyết các bài toán nhỏ của chúng.
- Kết hợp các kết quả của các bài toán con nhằm tạo ra nghiệm cho bài toán tổng thể.

Thông thường các kỹ thuật cụ thể để giải quyết các bài toán con (bước thứ 3) được định nghĩa rõ ràng và nghiêm ngặt, là thuật toán (bản chất định lượng). *Ví dụ:* kỹ thuật để mô phỏng bộ lọc tuyến tính được biểu diễn bởi hàm truyền đạt sử dụng phương pháp FIR, là tổng chập hoàn toàn xác định. Mặt khác, *"phương pháp luận"* toàn diện được sử dụng để ánh xạ bài toán thiết kế hoặc ước tính hiệu năng thành mô hình mô phỏng phù hợp và lựa chọn tập các kỹ thuật kiên định, tương thích để áp dụng vào mô hình đó, sẽ cần phải có các thủ thuật khám phá và những *"mánh khoé nghề nghiệp"*.

Mục đích cơ bản của hệ thống truyền thông là xử lý dạng sóng và ký hiệu, vì vậy mô phỏng hệ thống truyền thông là tìm cách ***phỏng tạo*** quá trình này bằng cách tạo và xử lý các giá trị mẫu của những dạng sóng này. Theo đó cần phải: *(i)* mô hình hóa các hoạt động xử lý tín hiệu (được thực hiện bởi các khối chức năng trong hệ thống); *(ii)* tạo các dạng sóng đầu vào để thâm nhập vào các điểm khác nhau của hệ thống; *(iii)* điều khiển các mô hình bằng các dạng sóng đầu vào phù hợp để tạo các dạng sóng đầu ra (là đầu vào của khối chức năng khác); *(iv)* phân tích những dạng sóng này để tối ưu các tham số thiết kế, hoặc để đạt được các phép đo hiệu năng.

Để minh hoạ các khía cạnh của phương pháp luận, ta sử dụng hệ thống truyền thông số làm việc trên kênh truyền thông di động "*thay đổi theo thời gian*". Kênh này gây *méo tuyến tính*, có thể giảm thiểu méo này bằng bộ cân bằng trong máy thu. Các giải pháp thiết kế chi tiết bộ cân bằng sẽ được dùng để minh hoạ một số khía cạnh của phương pháp luận. Thuộc tính thay đổi theo thời gian của kênh (do tính di động) làm cho tín hiệu thu thay đổi ngẫu nhiên, và là hàm của thời gian. Thay đổi ngẫu nhiên trong tín hiệu thu gọi là pha đinh. Khi công suất tín hiệu thu dưới mức ngưỡng, thì hiệu năng của hệ thống được đánh giá bởi xác suất lỗi không chấp nhận được và hệ thống sẽ thông báo là không phục vụ. *Xác suất ngừng hoạt động* của hệ thống được định nghĩa là phần trăm thời gian mà hệ thống truyền thông *"không khả dụng"* (do điều kiện kênh tồi gây ra tỷ số lỗi vượt quá giá trị ngưỡng quy định). Để ước tính xác suất ngừng hoạt động cần phải mô phỏng hệ thống dưới nhiều điều kiện kênh (kịch bản kênh), là một nhiệm vụ cần nhiều tính toán. Theo đó, ta sẽ thảo luận các phương pháp giảm thiểu tính toán trong mô phỏng. Muốn vậy, ta tiến hành ba bước cơ bản sau:

Trước hết, là lập mô hình và xác định tham số. Giải pháp tổng thể để mô phỏng mức dạng sóng hệ thống truyền thông khá dễ hiểu. Ta bắt đầu bằng việc mô tả phân chia hệ thống để mô phỏng hệ thống ở dạng sơ đồ khối, trong đó mỗi khối chức năng thực hiện một hoạt động xử lý tín hiệu cụ thể. Mô hình mô phỏng cho mỗi khối chức năng được chọn từ thư viện các mô hình khả dụng, và tạo mô hình sơ đồ khối bằng cách kết nối tập các khối được chọn. Trước khi thực hiện mô phỏng, cần phải xác định rõ các giá trị cụ thể hoặc khoảng các giá trị được phép cho các tham số của mỗi khối (chẳng hạn như băng thông của bộ lọc). Đơn giản hóa sơ đồ khối đến mức có thể và chia nhỏ nếu cần thiết. Việc ánh xạ bài toán thiết kế và/hoặc ước tính hiệu năng thành mô hình mô phỏng là một trong những bước khó nhất trong phương pháp luận. Thời gian mô phỏng và độ chính xác của kết quả mô phỏng phụ thuộc vào cách thực hiện điều này.

Sau đó, là thực hiện mô phỏng. Thực hiện mô phỏng bao gồm tạo các giá trị mẫu của tất cả các dạng sóng đầu vào hoặc tác nhân kích thích điều khiển mô hình mô phỏng. Các tín hiệu, tạp âm, nhiễu được biểu diễn bởi các quá trình ngẫu nhiên, và sử dụng các bộ tạo số ngẫu nhiên để tạo các giá trị mẫu của các quá trình ngẫu nhiên. Trong mô phỏng, các đầu ra của các bộ tạo số ngẫu nhiên được đưa vào các khối để: *(i) điều khiển* mô hình mô phỏng; *(ii) tạo* các giá trị mẫu tại các đầu ra của các khối chức năng. Một số mẫu đầu ra được ghi lại và được phân tích trong khi đang thực hiện mô phỏng *hoặc* tại thời điểm kết thúc mô phỏng, *ước tính* các phép đo hiệu năng như: tỉ số tín hiệu trên tạp âm SNR, sai số trung bình bình phương MSE, xác suất lỗi.

Cuối cùng, là xác định tính hợp lệ. Cũng là bước rất quan trọng trong mô phỏng, ta xác định tính hợp lý của các kết quả mô phỏng bằng cách dùng các xấp xỉ giải tích, các giới hạn

hoặc các kết quả đo khả dụng. Các kết quả đo cụ thể chỉ khả dụng đối với thời điểm kết thúc chu kỳ thiết kế sau khi nguyên mẫu được xây dựng. Thậm chí, khi hệ thống nguyên mẫu là khả dụng nhưng chỉ thực hiện hữu hạn các phép đo. Bản chất tốn kém của đo kiểm là nhân tố chính dẫn đến thực hiện mô phỏng. Tuy nhiên, một khi tính hợp lý (phê chuẩn) trái ngược với các kết quả đo, khi này ta dùng phương pháp luận để: *(i)* kiểm tra lại các mô hình; *(ii)* thiết lập tính khả tin của các kết quả mô phỏng.

Hệ thống truyền thông thực tế luôn quá phức tạp để mô hình hóa và mô phỏng cho dù tài nguyên tính toán khả dụng không giới hạn đi chăng nữa. Vì vậy, cần phải dùng nhiều kỹ thuật để giảm toàn bộ tính phức tạp của bài toán mô phỏng thành một bài toán trong phạm vi kiểm soát của: tài nguyên máy tính khả dụng, thời gian khả dụng, độ chính xác mong muốn. Các kỹ thuật, *kỹ năng nghề nghiệp*, được coi là phương pháp luận và được mô tả trong các phần sau ở dạng các ví dụ minh họa.

Sẽ được thấy rõ, dù điều khiển thời gian hay điều khiển sự kiện, mô phỏng phải được tổ chức và triển khai phù hợp nếu muốn có được kết quả khả tin và khả kiểm. Các khái niệm được trình bày có tính định hướng nghiên cứu cho các chương sau. Cấu trúc của mô phỏng thường phản ánh giải pháp được dùng để thiết kế hệ thống thực tế. Tuy nhiên, nhiều khéo léo nghề nghiệp sẽ được thảo luận, có thể ứng dụng vào mô phỏng để đảm bảo các kết quả mô phỏng phản ánh chính xác hoạt động của hệ thống theo thiết kế hoặc ước lượng.

2.2. Các mặt của phương pháp luận

Giải pháp toàn diện hay phương pháp luận để giải quyết bài toán thiết kế hoặc ước tính hiệu năng phụ thuộc vào bản chất bài toán cụ thể. Trong khi khó để trình bày phương pháp luận như tập các quy tắc hay thuật toán độc lập thì tồn tại một số khía cạnh chung của phương pháp luận có thể áp dụng cho nhiều bài toán mô phỏng khác nhau. Trước hết, ta mô tả chúng và sau đó trình bày tập các phương pháp cụ thể để giải quyết tập các bài toán riêng biệt.

2.2.1. Ánh xạ bài toán thành mô hình mô phỏng

Điểm bắt đầu của mô phỏng là trình bày tường minh bài toán và mục đích của mô phỏng. Để minh họa các mặt khác nhau của phương pháp luận, ta dùng hệ thống thông tin di động làm ví dụ và xét 2 vấn đề sau:

• ***Thiết kế bộ cân bằng***: Xác định số nhánh, khoảng cách nhánh và số bit được sử dụng để thực hiện các phép toán số học trong bộ cân bằng máy thu.

• ***Ước lượng hiệu năng hệ thống***: Xác định E_b/N_0 cần thiết để duy trì hiệu năng có thể chấp nhận được (mô tả chi tiết hơn về hệ thống và các đặc tính kỹ thuật hiệu năng của nó được đề cập ở phần sau của chương).

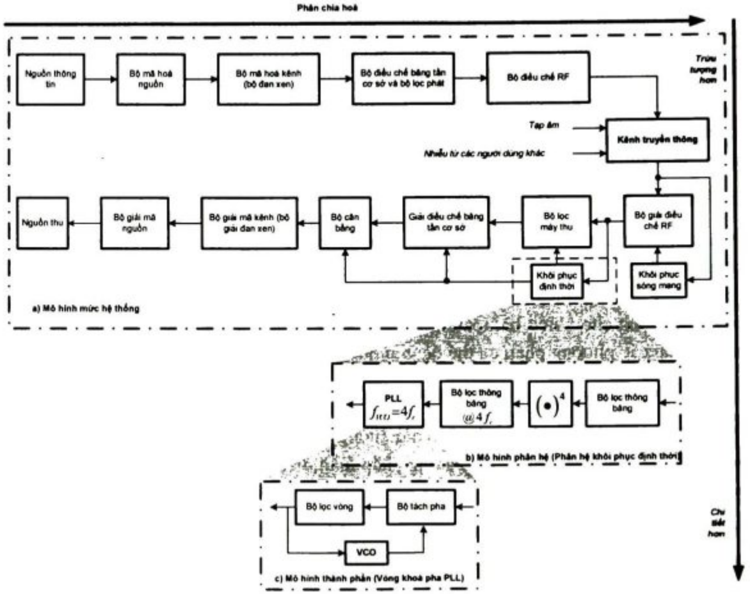
Bài toán thứ nhất bàn về thiết kế chi tiết một thành phần máy thu, bài toán thứ hai thuộc về ước tính hiệu năng mức hệ thống. Hai bài toán này cần có: các giải pháp ở dạng một phần của hệ thống để lập mô hình, mức độ chi tiết của mô hình, kỹ thuật lập mô hình, kỹ thuật mô phỏng, thủ tục ước tính. Ngoài ra, bài toán thứ nhất được giải quyết trước khi tiến tới bài toán thứ hai.

Dù ta đang giải quyết bài toán thiết kế chi tiết hay ước tính hiệu năng mức hệ thống thì điểm bắt đầu luôn là sơ đồ khối chi tiết, trình bày phần hệ thống cần được mô phỏng. Trình bày sơ đồ khối khởi đầu thường chi tiết hơn mức cần thiết. Tuy nhiên, là thói quen và hữu hiệu vì tại điểm bắt đầu bao gồm *"mọi thứ mà ta có thể nghĩ về"* sơ đồ khối toàn diện khởi đầu.

Mô hình mô phỏng cuối cùng được tạo ra từ sơ đồ khối khởi đầu được đơn giản hóa. Ứng dụng ba lớp kỹ thuật sau để tạo lập mô hình mô phỏng:

- Trình bày phân cấp
- Phân chia hóa và điều kiện hóa
- Đơn giản hoá (xấp xỉ, giả định).

Trình bày phân cấp



Hình 2.1: Minh họa quá trình mô hình hóa hệ thống

Phân cấp là giải pháp thường dùng để giảm tính phức tạp trong mô hình hóa, thiết kế phần mềm và các ứng dụng khác. Trong viễn cảnh hệ thống truyền thông, phân cấp được sử dụng để: *(i)* quản lý và giảm tính phức tạp của mô hình mô phỏng; *(ii)* giảm tải tính toán trong quá trình mô phỏng mô hình. Trình bày phân cấp được thực hiện trong các *"lớp"* khác nhau bắt đầu từ mô hình mức *"hệ thống"* và qua các lớp phân hệ, lớp thành phần và lớp vật lý. Minh họa các *"lớp"* cho một hệ thống truyền thông cụ thể được cho ở hình 2.1. Số các lớp, thuật ngữ được

sử dụng để định nghĩa một lớp cho trước là không duy nhất. Số lượng lớp trong hệ thống có thể là tùy ý, những gì được thấy là phân hệ trong một viễn cảnh được xét là một hệ thống trong viễn cảnh khác. Tuy nhiên, ta dùng thuật ngữ hệ thống để nói về toàn bộ thực thể quan tâm. Ở dạng phần cứng, hệ thống thường được xem là những gì chứa trong giá, tủ và hộp. Hệ thống chứa các phân hệ (thường được thực thi tại mức bo mạch).

Trong trình bày phân cấp hoặc lập mô hình hệ thống, các khối ở các lớp thấp hơn trong phân cấp sẽ chi tiết hơn, ngược lại các khối tại các lớp cao hơn trừu tượng hơn và thuộc về chức năng tổng thể của khối. Phân giải thành các lớp thấp hơn cho đến khi không thể. Mức thấp nhất thường dựa trên các thành phần như điện trở, tụ điện và vi mạch.

Trong toàn cảnh hệ thống truyền thông, mô hình mức hệ thống được cho ở hình 2.1(a) gồm các khối chức năng như: nguồn thông tin, bộ mã hoá, bộ giải mã, bộ điều chế, bộ giải điều chế, bộ lọc và kênh. Mỗi khối chức năng là một phân hệ và được phân giải (khai triển sâu hơn) để được chi tiết hơn. Ví dụ phân hệ khôi phục định thời được phân tích thành: hai bộ lọc thông băng, bộ phi tuyến bậc 4 và vòng khoá pha PLL như ở hình 2.1(b). Phân tích hơn nữa nhận được mô hình mức *"thành phần"*. Ví dụ, bộ lọc thông băng được cho ở hình 2.1(b) có thể là bộ lọc tương tự, bộ lọc siêu cao hay bộ lọc số. Trường hợp bộ lọc tương tự, có thể khai triển thành các mô hình mức *"mạch điện"*. Trường hợp bộ lọc số, việc phân tích sẽ đi xuống mức bit như các bộ cộng, bộ nhân, bộ tích luỹ. Dưới lớp này chứa các transistor và các cổng logic cụ thể. Tuy nhiên, trong viễn cảnh mô phỏng mức dạng sóng ta rất hiếm khi đi xuống mức chi tiết này. Mô hình lớp *"thành phần"* cho PLL được minh họa ở hình 2.1(c).

Ba lý do chính để dùng khái niệm phân cấp là: *(i)* Quản lý tính phức tạp của mô hình mô phỏng; *(ii)* Giảm tải tính toán khi mô phỏng mô hình. Tổng quát, nên thực hiện mô phỏng ở mức trừu tượng cao nhất có thể, phù hợp với mục đích mô phỏng bởi lẽ mức trừu tượng càng cao thì càng ít tham số hơn và mô phỏng hiệu quả hơn. Chẳng hạn khi thiết kế bộ cân bằng, nó được mô phỏng ở mức bit, trong khi đó kênh được mô phỏng ở mức trừu tượng hơn nhiều (dùng hàm truyền đạt để biểu diễn kênh). Tương tự, bộ lọc băng tần cơ sở số trong máy thu không nhất thiết phải mô phỏng ở mức bit nếu mục đích của mô phỏng là ước lượng hiệu năng hệ thống. Tất nhiên, cách thực thi bộ lọc sẽ không ảnh hưởng đến hiệu năng của toàn bộ hệ thống miễn là duy trì hàm truyền đạt của bộ lọc; *(iii)* Liên quan đến phê chuẩn mô hình, ngoài việc giảm tính phức tạp, thời gian mô phỏng, các mô hình mức cao hơn sẽ có ít tham số hơn và cũng có thể dễ phê chuẩn hơn. Càng ít tham số hơn thì càng ít phép đo để đặc tính hóa mô hình hơn. Ví dụ mô hình mức mạch điện của bộ lọc Butterworth chứa nhiều giá trị thành phần hơn. Tuy nhiên, hàm truyền đạt mức cao của cùng bộ lọc Butterworth được đặc trưng chỉ bởi 2 tham số (bậc và băng thông), chúng đều dễ đo. Hơn nữa, khi mô hình mô phỏng ở mức trừu tượng cao hơn, thì sự phê chuẩn của các kết quả mô phỏng là đơn giản hơn, cần ít phép đo hơn.

Tại mức hệ thống, mô phỏng được thực hiện ở mức trừu tượng cao nhất sử dụng các mô hình "*tính cách*" (các hàm truyền đạt) chứ không phải là các mô hình vật lý. Dạng chức năng của mô hình tính cách thường được giả định hoặc từ các phép đo. Ví dụ, bộ lọc số có thể được mô phỏng tại mức bit và bộ lọc tương tự có thể được mô phỏng tại mức mạch. Mô hình mức cao hơn cho cả hai bộ lọc có thể được rút ra từ các mô phỏng mức bit hoặc mức mạch ở dạng hàm truyền đạt. Chỉ khi dùng mô hình bộ lọc ở mức cao hơn (là mô hình hàm truyền đạt) mang

lại hiệu quả tính toán cao. Tính chi tiết của mô hình mức thấp hơn (dù là bộ lọc tương tự hay bộ lọc số) đều bị ẩn hoàn toàn từ lớp cao hơn. Giải pháp tạo mô hình mức cao hơn này từ các tính chi tiết của mô hình mức thấp hơn, và thay thế ngược lại tại mức cao hơn được gọi là "chú giải *ngược*".

Trong giai đoạn đầu của chu kỳ thiết kế, hàm truyền đạt bộ lọc được giả định hoặc *"định rõ"* và nhận được đặc điểm hàm truyền đạt sau khi thiết kế và mô phỏng. Ở chu kỳ thiết kế sau, một khi bộ lọc được xây dựng, thì hàm truyền đạt của nó được đo và được dùng trong các mô phỏng mức cao hơn. Ngoài ra, phân cấp cũng làm giảm toàn bộ tính phức tạp mô hình và tải tính toán kết quả.

Cũng như quá trình mô hình hoá, việc thiết kế hệ thống truyền thông thực tế cũng theo thứ tự từ đỉnh qua các lớp. Trong quá trình thiết kế, các đặc tính kỹ thuật đi xuống qua các lớp của phân cấp và việc đặc trưng hóa (được đo hoặc được mô phỏng ở mức thấp hơn) theo dự phòng qua các lớp của phân cấp. Trong một số ứng dụng, có thể phải sử dụng các mức chi tiết khác nhau. Ví dụ khi thiết kế bộ cân bằng cần phải ước tính xác suất lỗi hệ thống là hàm của số bit được sử dụng cho bộ cân bằng. Trường hợp tất cả các phần của hệ thống quanh bộ cân bằng sẽ được mô phỏng tại mức trừu tượng rất cao trong khi chính bộ cân bằng được mô phỏng chi tiết hơn nhiều sử dụng một bộ mô phỏng khác. Giải pháp này thường gọi là "*đồng mô phỏng*".

Phân chia hóa và điều kiện hóa

Phân chia hóa: Phân chia hóa bài toán phức tạp thành một tập các bài toán nhưng độc lập nhau, được giải quyết riêng biệt và kết hợp các nghiệm, là một kỹ thuật hữu hiệu khác làm giảm tính phức tạp và tải tính toán. Trong khi phân cấp bàn về mức độ trừu tượng thì phân chia hóa cũng bàn về tính trừu tượng nhưng ở các khía cạnh khác của bài toán đó là: mô phỏng riêng biệt và kết hợp các kết quả. Vì vậy, phân chia hóa cho phép quan sát và kiểm tra sơ đồ khối theo *"chiều ngang"* trong khi phân cấp cho phép xét theo "*chiều đứng*". Trong viễn cảnh được cho ở hình 2.1 có thể tách biệt phần đồng bộ hóa, mã hoá ra khỏi bài toán và mô phỏng chúng riêng biệt.

Điều kiện hóa: Điều kiện hóa là một kỹ thuật khác rất giống với phân chia hóa, đơn giản ta cố định điều kiện hay trạng thái của một phần hệ thống và mô phỏng phần còn lại của hệ thống dưới các giá trị khác nhau của các biến điều kiện hay trạng thái. Mô phỏng riêng cho phần hệ thống được điều kiện hóa và các kết quả đạt được trong phần đầu được lấy trung bình theo phân bố của biến điều kiện đạt được trong phần thứ hai. Quá trình này được minh họa tốt nhất bằng một ví dụ.

Giả sử ta muốn ước tính xác suất lỗi hệ thống được cho ở hình 2.1 khi đồng bộ hóa (khôi phục sóng mang và định thời) không lý tưởng. Ta sử dụng phân chia hóa và điều kiện hóa để đơn giản hóa bài toán bằng cách ước tính xác suất lỗi có điều kiện trong hệ thống với các giá trị khác nhau của lỗi pha sóng mang và định thời, sau đó mô phỏng hệ thống đồng bộ để đạt được phân bố của các lỗi định thời. Sau đó lấy trung bình xác suất lỗi có điều kiện theo phân bố lỗi định thời và lỗi pha. Những gì ta đang làm ở đây là phép toán trong thống kê liên quan các giá trị kỳ vọng có điều kiện. Tổng quát:

$$\begin{aligned} E_{XY}\{g(X,Y)\} &= \iint g(x,y) f_{XY}(x,y)\,dxdy \\ &= \int \underbrace{\left\{ \int g(x,y) f_{X|Y}(x|y)\,dx \right\}}\, f_Y(y)\,dy \end{aligned} \tag{2.1}$$

Ở dạng kỳ vọng toán học có điều kiện là:

$$E_{XY}\{g(X,Y)\} = E_Y\left\{\underbrace{E_{X|Y}\left[g(X,Y)\right]}_{\text{Kỳ vọng có điều kiện}}\right\} \tag{2.2}$$

Trở lại ví dụ, xác định BER khi có lỗi định thời và lỗi pha, áp dụng nguyên lý này ta có:

$$\hat{P}_E = \iint \hat{P}r\{\text{lỗi} \mid \tau,\theta\}\, \hat{f}_{T\Theta}(\tau,\theta)\,d\tau d\theta \tag{2.3}$$

Trong đó $\left\{\hat{P}r\{\text{lỗi} \mid \tau,\theta\}\right.$ là ước tính xác suất lỗi có điều kiện dựa trên mô phỏng với giả thiết có lỗi pha θ và lỗi định thời τ. Kết quả của trung bình hóa $\hat{P}_E$ là xác suất không điều kiện (toàn bộ) của lỗi, $f_{T\Theta}(\tau,\theta)$ là phân bố được ước tính (mô phỏng) của lỗi pha và lỗi định thời được tạo ra bởi hệ thống đồng bộ. Chú ý rằng hệ thống đồng bộ được mô phỏng bởi chính nó, các kết quả được lấy trung bình. Điều này dẫn đến mô phỏng hai hệ thống đơn giản hơn và thời gian mô phỏng ít hơn.

Nếu coi rằng các hệ thống khôi phục pha và định thời tạo ra các lỗi định thời và các lỗi pha *độc lập* thì các phần này có thể được phân chia hóa và được mô phỏng riêng biệt để đạt được các ước tính phân bố của lỗi định thời $\hat{f}_T(\tau)$ và lỗi pha $\hat{f}_\Theta(\theta)$. Nhận được phân bố đồng thời của lỗi pha và lỗi định thời là:

$$\hat{f}_{T\Theta}(\tau,\theta) = \hat{f}_T(\tau)\hat{f}_\Theta(\theta) \tag{2.4}$$

sau đó thay vào (2.3) để lấy trung bình.

Cần chú ý rằng, phân chia hóa bàn về việc phân tách bài toán, và điều kiện hóa định hướng phân chia hóa và quan trọng hơn là giúp tích hợp các kết quả. Giả định độc lập, phù hợp cũng hỗ trợ trong quan hệ hóa các kết quả. Phần sau sẽ là trường hợp các phần mô phỏng tạo ra hiện tượng độc lập thống kê và các quá trình cần được kết hợp.

Đơn giản hóa và xấp xỉ hóa

Tại điểm bắt đầu mô hình sơ đồ khối ban đầu thường được chi tiết hóa tới mức có thể. Tính phức tạp của mô hình tổng thể và các mô hình phân hệ được giảm bằng việc loại bỏ các khối ít ảnh hưởng lên bài toán, xấp xỉ hóa và đơn giản hoá bằng cách kết hợp các khối.

Ví dụ về cách rút gọn sơ đồ khối, xét bài toán ước tính hiệu năng mức hệ thống. Nếu coi kênh là kênh thay đổi rất chậm theo thời gian và hệ thống đang hoạt động tại SNR cao thì có

thể cho rằng các lỗi đồng bộ sẽ rất nhỏ và vì vậy có thể bỏ qua ảnh hưởng của đồng bộ trong quá trình ước tính hiệu năng. Khi này không cần mô phỏng các phần khôi phục định thời và khôi phục sóng mang và có thể xoá bỏ ra khỏi sơ đồ khối.

Mở rộng việc dùng xấp xỉ hóa và giả định để đơn giản mô hình mô phỏng. Giả định vả xấp xỉ hóa được sử dụng phổ biến nhất bao gồm việc tuyến tính hóa và bất biến hóa theo thời gian. Hầu hết các hệ thống thực tế, khi được quan trắc trong thời gian dài và trên dải tín hiệu đầu vào thay đổi rộng, có thể biểu lộ tính phi tuyến và thay đổi theo thời gian nhưng chúng có thể được xấp xỉ tốt bởi các mô hình tuyến tính và bất biến theo thời gian trong các khoảng thời gian ngắn và các mức tín hiệu thấp.

Bất biến theo thời gian ngụ ý rằng trong khoảng thời gian mô phỏng, các đặc tính của tín hiệu và các thành phần hệ thống đang được mô phỏng là không thay đổi. Thực tế, khái niệm bất biến theo thời gian là tương đối. Nếu tham số hệ thống thay đổi chậm thì có thể coi một số tình huống là cố định trong khoảng thời gian mô phỏng. Chẳng hạn, xét bài toán ước tính BER trên kênh vô tuyến trong đó các ăng ten phát và thu là dừng. *Nếu: (i)* những thay đổi trong đặc tính kênh là do thay đổi điều kiện khí quyển; *(ii)* tốc độ ký hiệu truyền dẫn là vài triệu ký hiệu trên giây thì có thể coi kênh là *"tựa tĩnh"*. Điều đó nói rằng kênh vẫn duy trì cùng một điều kiện trong khi vài trăm triệu ký hiệu đi qua nó và đặc biệt có ý nghĩa khi xác định xác suất lỗi tức thời trong điều kiện kênh cho trước. Nếu BER được ước tính là khoảng 10^{-3} thì chỉ cần mô phỏng vài nghìn ký hiệu để ước tính BER. Điều này thể hiện khoảng thời gian mô phỏng là vài ms trong khi hằng số thời gian của kênh là khoảng vài phút vì vậy nó là hợp lý để coi kênh là tĩnh trong khoảng thời gian mô phỏng. Việc lấy xấp xỉ tựa tĩnh có vai trò rất quan trọng để đơn giản hoá các mô hình mô phỏng.

Giả định tựa tĩnh và đơn giản hoá có thể được áp dụng cho hệ thống bất kỳ trong đó tồn tại hiện tượng và các quá trình có các độ rộng băng khác nhau đáng kể. Khi này, có thể mô phỏng các ảnh hưởng của quá trình thay đổi nhanh hơn trong khi đó vẫn đang coi quá trình chậm trong trạng thái cố định. *Vì vậy, có thể coi giả định tựa tĩnh là yêu cầu để phân chia hóa và điều kiện hóa.*

Tương tự, ta sử dụng các phép xấp xỉ tuyến tính cho các thành phần phi tuyến. Tổng quát, việc phân tích các mô hình phi tuyến là rất phức tạp nhưng với mô phỏng có phần dễ dàng hơn nhưng chúng vẫn đặt ra vài vấn đề. Mỗi khi có thể ta nên cố gắng lấy xấp xỉ thuộc tính của các thành phần này bằng các mô hình tuyến tính.

Cuối cùng, ta sử dụng nhiều nguyên lý hệ thống tuyến tính để đơn giản hóa sơ đồ khối. Ta có thể kết hợp vài khối nối tiếp và song song thành một khối bằng cách nhân hoặc cộng các hàm truyền đạt. Trường hợp các khối tuyến tính bất biến cho phép thay đổi thứ tự các khối để đơn giản mô hình đơn. Các loại đơn giản đặc biệt có ý nghĩa khi mô phỏng ước tính hiệu năng bởi lẽ: *(i)* các mô phỏng ước tính hiệu năng thường rất lâu; *(ii)* khác với trường hợp mô phỏng được dùng để hỗ trợ thiết kế chi tiết, ta không quan tâm quan sát tiến hóa và tiến trình của dạng sóng qua mỗi khối chức năng mà chỉ quan tâm so sánh các dạng sóng vào/ra và đếm các lỗi. Theo đó, các dạng sóng trung gian rất ít được quan tâm hoặc sử dụng. Vì vậy, có thể rút gọn toàn bộ hệ thống còn rất ít khối dẫn đến giảm đáng kể thời gian mô phỏng. Nếu các khối có cùng mức độ phức tạp thì kết hợp hàm truyền đạt của n khối sẽ dẫn tới tiết kiệm tính toán khoảng n lần.

2.2.2. Mô hình hóa các khối chức năng

Vai trò của mỗi khối chức năng trong hệ thống truyền thông là thực hiện chức năng xử lý tín hiệu cụ thể, vì vậy mô hình mô phỏng sẽ phản ánh chức năng này với các mức trừu tượng thay đổi. Bất chấp tính chi tiết cục bộ, mô hình mô phỏng tiếp nhận chuỗi mẫu dạng sóng đầu vào theo trình tự thời gian và tạo ra tập các mẫu đầu ra cũng theo trình tự thời gian ứng với đặc tính truyền đạt cụ thể. Một số lựa chọn và xem xét phải tính đến khi xây dựng mô hình và ta mô tả một số vấn đề phương pháp luận liên quan đến việc mô hình hoá trong các phần sau.

Mô hình mô phỏng của phân hệ hoặc thành phần (khối) là biến đổi của dạng:

$$\left\{y[k], y[k-1], ..., y[k-m]\right\} = \text{F}\left\{x[k-j], x[k-j-1], ..., x[k-j-n]; k; p_1, p_2, ..., p_q\right\} \quad (2.5)$$

Trong đó $x[k]$ biểu diễn các mẫu đầu vào, $y[k]$ biểu diễn các mẫu đầu ra, $p_1, p_2, ... p_q$ biểu diễn các tham số của khối và $k = m, 2m, 3m, ...$ là chỉ số thời gian. Mô hình sử dụng n mẫu đầu vào để tạo m mẫu đầu ra trên "*dẫn chứng*" của mô hình theo biến đổi F, sẽ được định nghĩa theo: các mẫu đầu vào, các tham số của khối, chỉ số thời gian k. Nếu biến đổi F không phụ thuộc vào chỉ số k thì mô hình là bất biến theo thời gian. Nếu $m > 0$ thì mô hình được xét là mô hình vào/ra khối và khi $m = 0$ ta có mô hình từng mẫu. Nếu $n = 0$ thì mô hình không nhớ.

Khi xây dựng mô hình cho khối chức năng và thực hiện mô phỏng phải xét đến nhiều nhân tố. Những nhân tố này liên quan với nhau thậm chí chúng xuất hiện bất kỳ.

Biểu diễn tương đương thông thấp

Các hệ thống truyền thông chứa thành phần và tín hiệu có bản chất thông dải hoặc thông thấp. Từ viễn cảnh mô phỏng, nhận được các thuận lợi tính toán nếu biểu diễn tất cả tín hiệu và phần tử hệ thống ở dạng tương đương thông thấp phức. Với tín hiệu và hệ thống tuyến tính, tương đương thông thấp có được bằng cách dịch phổ thông băng từ tần số sóng mang về tần số $f = 0$ và mô hình tuyến tính của khối có thể được thực hiện bằng cách sử dụng biểu diễn tương đương thông thấp của các tín hiệu đầu vào/ra và biến đổi tín hiệu, được chi tiết hóa ở chương 4.

Tương đương thông thấp của tín hiệu tất định đạt được qua chuyển dịch phổ tần của nó, trong khi đó đối với tín hiệu ngẫu nhiên thực hiện dịch mật độ phổ công suất. Nếu phổ thông băng không đối xứng qua các sóng mang thì biểu diễn tương đương thông thấp trong miền thời gian là giá trị phức. Hơn nữa, khi này các thành phần của quá trình ngẫu nhiên tương đương thông thấp sẽ được tương quan nhau.

Đối với loại hệ thống phi tuyến cũng có thể sử dụng biểu diễn tương đương thông thấp. Chi tiết hóa mô hình tương đương thông thấp cho hệ thống phi tuyến được mô tả ở chương 12.

Lấy mẫu

Một khi tín hiệu và hệ thống là thông thấp (trong trường hợp thông băng của chúng đư biểu diễn bởi tương đương thông thấp) thì chúng có thể được lấy mẫu và biểu diễn bởi các mẫu cách đều nhau. Trường hợp thông thấp lý tưởng, tốc độ lấy mẫu nhỏ nhất phải gấp hai lần độ rộng băng tần của tín hiệu. Tuy nhiên, thực tế các hàm tần số có thể không được giới hạn độ rộng băng thì tốc độ lấy mẫu thường lấy là 8 đến 16 lần độ rộng băng tần. Trường hợp các hệ thống số thì tốc độ lấy mẫu thường chọn là 8 đến 16 lần tốc độ ký hiệu. Các nhân tố như lỗi

chồng phổ, méo tần số khi thực thi bộ lọc và nở rộng băng thông do tính phi tuyến đều phải được tính đến khi chọn tốc độ lấy mẫu. Có thể giảm thiểu các ảnh hưởng này bằng cách tăng tốc độ lấy mẫu nhưng lại làm tăng tải tính toán (thời gian mô phỏng lâu hơn) vì vậy cần phải dung hoà độ chính xác và thời gian mô phỏng. Lấy mẫu đa tốc độ, kích thước bước khả biến... là các kỹ thuật được áp dụng để giảm tải tính toán.

Mô hình tuyến tính và phi tuyến

Trong khi hầu hết các khối là tuyến tính thì một phần đáng kể của hệ thống cần phải xử lý phi tuyến. Tồn tại một số xử lý phi tuyến là chủ ý và không chủ ý. Minh chứng cho loại chủ ý như: các bộ cân bằng hồi tiếp quyết định, tính phi tuyến trong các phân hệ đồng bộ... Với loại không chủ ý như: tính cách phi tuyến của các bộ khuếch đại.

Như xấp xỉ đầu tiên, có thể mô hình hóa hầu hết tính phi tuyến là có các ảnh hưởng tuyến tính lên các tín hiệu truyền thông, đặc biệt nếu tín hiệu là tín hiệu đường bao không đổi. Tuy nhiên, trong các hệ thống đa sóng mang hoặc trong các hệ thống đơn sóng mang với điều chế biên độ cầu phương bậc cao (M-QAM) thì tính cách phi tuyến không chủ ý có ảnh hưởng đáng kể lên hiệu năng hệ thống và vì vậy cần phải xét các mô hình mô phỏng phi tuyến. May thay, hầu hết tính cách phi tuyến này đều có thể được mô hình hóa một cách hiệu quả bằng cách sử dụng biểu diễn tương đương thông thấp phức.

Tồn tại nhiều giải pháp để mô hình hóa các hệ thống có tính phi tuyến. Chúng gồm, các phương pháp phi tuyến chuỗi lũy thừa không nhớ, các mô hình phi tuyến chọn lọc tần số có nhớ và các phương trình vi phân phi tuyến. Tổng quát, việc phân tích toán học các hệ thống phi tuyến và ước lượng các ảnh hưởng của tính phi tuyến là khó. Tuy nhiên, dùng mô phỏng thì đơn giản hơn ngay cả với các mô hình phi tuyến chọn lọc tần số.

Các mô hình phi tuyến được phân thành hai loại chính: *(i)* các mô hình khối vào-ra; *(ii)* các phương trình vi phân phi tuyến. Tập các mô hình thứ nhất thường dựa vào các phép đo trong khi đó lớp các mô hình thứ hai thường được rút ra từ việc mô hình hóa tính cách vật lý của thiết bị. Các giải pháp của các mô hình phương trình vi phân phi tuyến được thực hiện bằng cách sử dụng các mô hình tích phân bước thời gian khả biến (có hiệu quả tính toán nhất mặc dù chúng có thể mất nhiều thời gian thiết lập hơn). Cũng có thể phân giải một phân hệ phi tuyến vào dạng sơ đồ khối và mô phỏng dạng sơ đồ khối bằng cách sử dụng các khối cơ bản sẵn có trong thư viện mô phỏng. Giải pháp này mặc dù dễ thiết lập hơn nhưng sẽ không là giải pháp hiệu quả tính toán cao nhất.

Một nhân tố quan trọng phải được xét đến khi mô phỏng các phần tử phi tuyến là tính phi tuyến gây ra dãn rộng độ rộng băng tần (nở phổ) vì vậy tốc độ lấy mẫu phải được chọn đủ lớn để bắt giữ các ảnh hưởng của sự nở rộng độ rộng phổ tần.

Bất biến theo thời gian

Như đã được trình bày, tất cả các hệ thống, các thành phần và các quá trình sẽ biểu lộ tính cách thay đổi theo thời gian khi quan sát trong khoảng thời gian dài. Việc sử dụng hay không sử dụng mô hình thay đổi theo thời gian được định hướng bởi một số nhân tố. Xét ba trường hợp sau:

Trường hợp thứ 1: Trong nhiều ứng dụng như mô hình hóa và mô phỏng sợi quang, các đặc tính sợi quang có thể thay đổi rất ít trên tuổi thọ của hệ thống truyền thông và vì vậy chọn mô hình bất biến theo thời gian.

Trường hợp thứ 2: Tính cách thay đổi theo thời gian có thể đáng kể nhưng tốc độ thay đổi của chúng rất thấp so với độ rộng băng thông của các phần bất biến theo thời gian của hệ thống. Khi này, lấy *xấp xỉ tựa tĩnh* là hợp lệ, thực hiện mô phỏng bằng cách sử dụng những chớp ảnh cố định của các phần thay đổi theo thời gian, sau đó lấy trung bình các kết quả (nghĩa là, phân chia hóa và trung bình hóa). Trong hai trường hợp này, phép đo hiệu năng là một số trung bình hóa dài hạn chứ không phải là tính cách động.

Trường hợp thứ 3: Đôi khi được chứng thực là mô phỏng *động* các thay đổi theo thời gian, được dùng khi những biến đổi theo thời gian là *nhanh* và hiệu năng của hệ thống dựa vào tính cách nhất thời hay động của hệ thống. Trường hợp điển hình là việc bắt và bám tính cách nhất thời của phân hệ đồng bộ làm việc trong môi trường kênh pha đinh nhanh. Mô hình mô phỏng cho trường hợp này sẽ là mô hình đường trễ rẽ nhánh có các hệ số khuếch đại nhánh thay đổi theo thời gian, thường được mô hình hóa là các quá trình ngẫu nhiên được lọc.

Trong khi khá dễ dàng rút ra và thực thi mô hình mô phỏng đường trễ rẽ nhánh cho các hệ thống thay đổi theo thời gian thì hai nhân tố phải được tính đến là: *(i)* do tính thay đổi theo thời gian gây ra hiện tượng nở phổ tần đáng kể, hậu quả là phải lấy mẫu cao hơn; *(ii)* không được thay đổi thứ tự các khối có tính thay đổi theo thời gian bởi lẽ không tồn tại tính chất giao hoán đối với các hệ thống thay đổi theo thời gian.

Tính nhớ

Nếu đầu ra tức thời $y[k]$ của một phần tử phụ thuộc vào đầu vào tức thời $x[k]$ thì phần tử đó là không nhớ; ngược lại là có nhớ. Các bộ lọc, do có tính cách chọn lọc tần số nên thuộc loại có nhớ (lưu ý, tính cách chọn lọc tần số là đồng nghĩa với tính có nhớ). Ngoài ra, một số loại phi tuyến có nhớ và tồn tại nhiều mô hình khả dụng để mô phỏng chúng. Đặc biệt cẩn thận khi thực hiện các mô hình có nhớ theo lưu giữ các trạng thái nội tại của mô hình sao cho mô hình có thể trở lại. Ví dụ khi dùng mô hình bộ lọc tổng quát cho một vài trường hợp trong sơ đồ khối thì trạng thái bên trong của mỗi trường hợp của bộ lọc phải đưoc lưu giữ riêng biệt sao cho khi mô hình bộ lọc được dùng đến vài lần trong quá trình mô phỏng nó luôn được đi vào trạng thái nguyên vẹn trước đó.

Mô phỏng trong miền thời gian và miền tần số

Có thể mô hình hóa và mô phỏng quan hệ vào/ra của các khối chức năng trong miền thời gian hoặc miền tần số. Gánh nặng tính toán của 2 giải pháp cho các khối tuyến tính thường là tương đương, miền phù hợp để thực thi phụ thuộc vào miền mà các đặc tính kỹ thuật được cung cấp khởi đầu, chẳng hạn nếu bộ lọc được xác định theo đáp ứng tần số thì nên chọn miền tần số. Trường hợp phi tuyến, các đặc tính kỹ thuật và việc thực hiện đa phần được thực hiện trong miền thời gian.

Trong khi việc thực hiện mô hình có thể trong cả miền thời gian và tần số thì thực tế thường dùng các mẫu trong miền thời gian để biểu diễn tín hiệu vào/ra. Các mô hình trong miền tần số, ví dụ như dùng FFT để mô phỏng các bộ lọc cần phải lưu đệm nội bộ các mẫu đầu

vào miền thời gian, lấy biến đổi vectơ đầu vào đã được lưu trong bộ đệm theo cách xử lý trong miền tần số, FFT ngược, và lưu đệm tại đầu ra. Phải lưu đệm bởi lẽ thực hiện biến đổi là hoạt động xử lý khối dựa trên tập các mẫu chứ không phải là xử lý theo từng mẫu. Trong quá trình mô phỏng, các mẫu đầu vào và đầu ra có thể xuất/nhập bộ đệm theo từng mẫu tại một thời điểm hoặc theo các khối N mẫu.

Xử lý khối

Một mô hình có thể tiếp nhận và xử lý một mẫu tại một thời điểm hoặc một khối N mẫu miền thời gian trên một lần cần đến. Hiệu quả tính toán của 2 phương pháp sẽ phụ thuộc vào tính phức tạp và phần mào đầu liên quan với lần gọi của mô hình. Nếu mô hình có số các trạng thái nội bộ và các tham số là nhỏ hoặc nếu thông tin mào đầu để gọi (hay làm xuất hiện) mô hình là nhỏ so với các tính toán được thực hiện bên trong mô hình, thì việc gọi (làm xuất hiện) mô hình trên cơ sở từng mẫu là thuận tiện và hiệu quả. Khi thông tin mào đầu để gọi mô hình là lớn thì việc sử dụng khối hoặc giải pháp xử lý vectơ có (mô hình được gọi bằng một vectơ đầu vào kích cỡ *N*) cho ta hiệu quả tính toán cao.

Xử lý khối dẫn đến phải lưu đệm và giao diện tốt với các khối trước và theo khối. Xử lý khối gây trễ $N*T_s$ giây, bởi lẽ đầu ra không thể được tính toán cho đến khi tập hợp được toàn bộ N mẫu đầu vào và được chuyển vào mô hình. Kể cả mô hình vào/ra khối trong vòng hồi tiếp sẽ tạo ra các kết quả sai do trễ xử lý lớn. Ngoài ra, nếu mô hình xử lý khối được trộn lẫn với mô hình phi tuyến thì nên trở lại phương pháp xử lý theo từng mẫu đối với phần tử phi tuyến tính, vì gây nghẽn đầu vào và xử lý trên cơ sở từng khối dùng nguyên lý xếp chồng, xếp chồng không được áp dụng cho phần tử phi tuyến. Ví dụ tình huống này xảy ra khi tính phi tuyến xuất hiện giữa hai bộ lọc FFT kiểu phát sinh và chồng chập. Bộ lọc FFT phát sinh và chồng chập dựa vào nguyên lý tuyến tính. Kỹ thuật tính toán đáp ứng bộ lọc cho các khối không chồng chập của các mẫu đầu vào và cộng các đáp ứng tại đầu ra. Với phần tử phi tuyến sau bộ lọc, phương pháp xử lý khối không thể thực hiện được cho phần tử phi tuyến bởi lẽ xếp chồng không áp dụng được đối với tính phi tuyến. Để xử lý chính xác thực hiện theo trình tự sau: *(i)* đáp ứng ra do mỗi khối của các mẫu đầu vào phải được cộng tại đầu ra của bộ lọc đầu tiên; *(ii)* mô hình phi tuyến xử lý đầu ra đã cộng của bộ lọc đầu tiên trên cơ sở từng mẫu; *(iii)* đầu ra của phần tử phi tuyến được xử lý bởi bộ lọc thứ hai sử dụng chồng chập và phương pháp cộng của xử lý khối. Chi tiết hóa cho bài toàn này được xét ở chương 12.

Một yếu tố khác phải tính đến khi bàn về xử lý khối là lập lịch. Nếu các mô hình khác nhau trong hệ thống sử dụng các kích thước khối vào/ra khác nhau thì khung mô phỏng nên có khả năng lập lịch thứ tự và tần suất gọi (gọi các phân hệ) phù hợp.

Xử lý kích thước bước khả biến

Xử lý đa tốc độ được sử dụng trong mô phỏng nếu mô hình hệ thống chứa các quá trình và hiện tượng có băng thông khác nhau lớn. Với lấy mẫu đa tốc độ, mỗi tín hiệu được lấy mẫu và được xử lý tại tốc độ phù hợp với băng thông của nó, dẫn đến cải thiện đáng kể hiệu quả tính toán. Khi sử dụng lấy mẫu đa tốc độ, cần đến nội suy để giao tiếp các luồng mẫu với các tốc độ lấy mẫu khác nhau.

Xử lý kích thước bước khả biến cũng thường được sử dụng để cải thiện hiệu quả tính toán. Giải pháp này thường được sử dụng trong các thường trình tích phân số để giải các

phương trình vi phân tuyến tính và phi tuyến. Nếu các phương trình vi phân cơ bản và các nghiệm của chúng được giải chính xác thì giải pháp này sẽ giảm đáng kể tải tính toán. Khi dùng kích thước bước khả biến trong mô hình thì đầu ra phải được lưu đệm và được lấy mẫu lại nếu các khối sau sử dụng kích thước bước không đổi.

Tham số hóa

Một trong những lý do cơ bản sử dụng mô phỏng là tối ưu hóa thiết kế, trong hầu hết các trường hợp làm giảm việc tìm kiếm giá trị tối ưu của các tham số tới hạn như băng thông của bộ lọc thu, điểm hoạt động của bộ khuếch đại và số mức lượng tử được sử dụng trong máy thu. Muốn vậy, các mô hình phải được tham số hóa một cách phù hợp và các tham số thiết kế chủ đạo nên được thể hiện ra ngoài; nghĩa là các mô hình nên có "*điều chỉnh ngoài*" để điều chỉnh tham số thiết kế một cách lặp trong quá trình mô phỏng. Phải tính đến số lượng tham số cần phải có đối với một mình cho trước. Tổng quát, nên nhỏ nhất có thể, vì hệ thống truyền thông phức tạp cần có nhiều thành phần. Nếu mỗi thành phần có nhiều tham số phụ thì toàn bộ không gian tham số sẽ rất lớn. Khi này sẽ rất khó để tối ưu thiết kế dùng mô phỏng. Theo đó, khi số lượng các tham số càng ít thì việc đo các giá trị tham số và phê chuẩn càng dễ dàng hơn.

Giao tiếp giữa các khối chức năng

Cùng với giải pháp mô hình hóa và mô phỏng được sử dụng cho mỗi thành phần phụ thuộc vào bản chất của thành phần sẽ được mô hình hóa và mô phỏng, cũng cần phải tính đến giao tiếp giữa các khối chức năng. Vì sơ đồ khối hệ thống gồm một tập các khối được kết nối với nhau, phải đảm bảo tính kiên định và tính tương thích cho cả cơ cấu mô phỏng và/hoặc người dùng. Sẽ dễ dàng hơn nếu các mô hình của các khối riêng biệt được xây dựng có các giao diện được định nghĩa và dẫn chứng rõ ràng. Một số lý giải cho sự không phù hợp gồm: các miền xử lý khác nhau, các loại tín hiệu, kích thước khối, kích thước bước, lấy mẫu đa tốc độ, đặc tính kỹ thuật các tham số không kiên định trong các khối khác nhau và nhiều lý do khác đã được sơ qua ở các phần trước.

Nhiều khó khăn khi mô phỏng mô hình phức tạp mức hệ thống xuất phát từ sự không phù hợp này. Vì vậy cần phải cẩn thận khi công thức hóa mô hình mô phỏng tổng thể, lựa chọn các khối riêng biệt và các tham số của chúng trong viễn cảnh không chỉ đối với các mô hình riêng biệt mà còn đối với mô hình chung tổng thể.

2.2.3. Mô hình hóa và mô phỏng quá trình ngẫu nhiên

Giả sử có một mô hình hệ thống với mức trừu tượng cao nhất và tính phức tạp thấp nhất, quan tâm các mặt của phương pháp luận, ứng dụng vào việc mô hình hóa và tạo các dạng sóng đầu vào (tín hiệu, tạp âm và nhiễu) điều khiển mô hình mô phỏng. Vì mục đích cơ bản của mô phỏng mức dạng sóng là *phỏng tạo* các dạng sóng trong hệ thống và tính toán một số *phép đo* dạng sóng một cách trung thực, muốn vậy trước hết cần phải lập mô hình và mô phỏng các dạng sóng đầu vào hoặc tác nhân kích thích.

Trong hệ thống truyền thông, dạng sóng mang thông tin, tạp âm và nhiễu mang bản chất ngẫu nhiên và được mô hình hóa bằng các quá trình ngẫu nhiên. Dừng là giả định đa năng nhất vì nó có thể được biện minh trong nhiều trường hợp dựa vào bản chất của tín hiệu hoặc chuỗi đang được mô hình.

Các quá trình ngẫu nhiên dừng được đặc trưng hóa bởi các phân bố xác suất nhiều chiều, khó để xác định, và ở dạng tổng quát rất khó tạo các giá trị mẫu của một quá trình dừng có phân bố n chiều bất kỳ. Ngoại trừ quá trình Gausơ dừng, hoàn toàn được xác định bởi phân bố bậc 2 (các tham số của nó là hàm tự tương quan và trung bình). Đối với các quá trình không Gausơ, thực tế giới hạn các đặc tính kỹ thuật thành các phân bố bậc 2.

Các giá trị mẫu của các quá trình ngẫu nhiên được dùng để điều khiển mô phỏng là các chuỗi các số ngẫu nhiên được tạo ra bởi các bộ tạo số ngẫu nhiên. Các thuật toán để tạo các chuỗi ngẫu nhiên với phân bố bất kỳ (bậc một và bậc hai) và các hàm tương quan thường được thấy trong các chương sau. Dưới đây đề cập một số mặt của phương pháp luận để lập mô hình và tạo các giá trị lấy mẫu của quá trình ngẫu nhiên.

Xấp xỉ Gausơ

Xấp xỉ Gausơ bởi định lý giới hạn trung tâm, nếu có các biến ngẫu nhiên X_i là độc lập, thì $Y = X_1 + X_2 + ... + X_n$ tiến dần tới phân bố Gausơ khi n lớn. Vì vậy tạp âm ăng ten thu, được đóng góp bởi một số lượng lớn các nguồn, có thể được xấp xỉ bởi quá trình Gausơ. Tương tự nhiễu từ một số lượng lớn người dùng cũng có thể được xấp xỉ bởi quá trình Gausơ và vì vậy không nhất thiết tạo các tín hiệu riêng biệt từ một số lượng lớn người dùng và lấy tổng chúng lại.

Biểu diễn quá trình tương đương

Khái niệm thứ hai bàn về quan điểm biểu diễn quá trình tương đương, được phát biểu như sau: Giả sử cho một quá trình ngẫu nhiên *X(t)* đi qua n khối và xuất hiện tại đầu ra của khối thứ n là một quá trình *Y(t)*. Nếu bằng một số phương tiện (thông qua phân tích vùng, hoặc xấp xỉ hóa, hoặc bằng chính mô phỏng), có thể suy ra các đặc tính của quá trình *Y(t)* thì toàn bộ quá trình xử lý sau đó (sau khối thứ *n*) ta có thể đơn giản hoá bằng cách chỉ cần tìm một chuỗi (biểu diễn các giá trị mẫu của *Y(t)*) vì vậy bỏ qua việc tạo và xử lý các giá trị mẫu của *X(t)* thông qua n khối. Khi *X(t)* là Gausơ và các khối là *tuyến tính* thì có thể chỉ ra rằng *Y(t)* cũng là Gausơ. Các tham số của quá trình *Y(t)* được rút ra từ phân tích hoặc mô phỏng *X(t)* qua *n* khối. Đáng tiếc, sẽ khó rút ra các đặc tính của *Y(t)* theo phép giải tích khi *X(t)* là bất kỳ và các khối là phi tuyến. Khi này, có thể dùng mô phỏng để ước tính các thuộc tính của *Y(t)* và có thể dùng các thuộc tính được ước tính này để tạo quá trình tương đương.

Giải pháp này được dùng để biểu diễn tạp âm pha trong các hệ thống truyền thông cũng như Jitter pha và định thời do các phân hệ đồng bộ tạo ra. Giả định được dùng phổ biến nhất ở đây là quá trình Gausơ *dừng*. Đối với các quá trình tạp âm lối vào thì mật độ phổ công suất PSD được giả định là trắng. Với các quá trình khác thì PSD được giả sử là dạng kín cho trước như tỉ số đa thức theo f^2, trong trường hợp này quá trình có thể được tạo ra bằng cách lọc quá trình Gausơ trắng bằng bộ lọc có hàm truyền đạt. Trường hợp các hàm PSD bất kỳ, chẳng hạn như trường hợp PSD Doppler của các kênh pha đinh, ta có thể *hoặc* xấp xỉ hóa phổ bằng tỉ số đa thức theo f^2 và áp dụng phương pháp thừa số hóa phổ để có được hàm truyền đạt của bộ lọc hoặc làm phù hợp mô hình trung bình di chuyển tự hồi quy ARMA trực tiếp theo PSD để đạt được các hệ số của bộ lọc đệ quy mà sẽ tạo ra PSD mong muốn.

Các quá trình phi Gausơ với PSD bất kỳ là khó để tổng hợp và mô phỏng hơn. Một phương pháp xử lý trường hợp này mặc dù rất khó áp dụng song có thể được thấy trong chương 7.

Các quá trình nhanh và chậm

Tồn tại khá nhiều hiện tượng ngẫu nhiên có băng thông hay "*các hằng số thời gian*" khác nhau nhiều. Nếu băng thông của quá trình này khác nhiều so với băng thông của quá trình khác thì nên dùng một trong hai giải pháp sau để giảm thời gian mô phỏng:

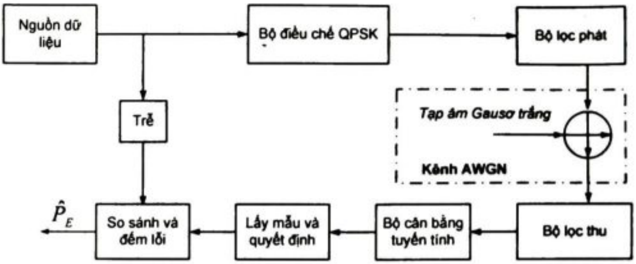
Giải pháp thứ nhất: Bài toán nên được phân chia hóa và điều kiện hóa thành quá trình chậm và thực hiện mô phỏng riêng biệt theo giá trị của quá trình chậm nếu có thể trong khi đó phân chia hóa hệ thống bàn về việc mô phỏng quá trình nhanh hơn. Theo đó, trong quá trình mô phỏng không cần phải tạo các giá trị mẫu của quá trình chậm bởi lẽ giá trị của nó thay đổi rất ít trong khoảng thời gian của số lượng lớn các mẫu của quá trình nhanh. Giải pháp này thường được sử dụng để mô phỏng hiệu năng của các hệ thống truyền thông đang hoạt động trên các kênh pha đinh chậm.

Giải pháp thứ hai: Lấy mẫu đa tốc độ, các quá trình được lấy mẫu tại các tốc độ khác nhau phù hợp với các băng thông của chúng sao cho số các mẫu được tạo ra trong khoảng thời gian mô phỏng là tỉ lệ nghịch với các băng thông của các quá trình tương ứng. Nếu cần có thể dùng phép nội suy và giảm mẫu để trộn các tín hiệu này với nhau tại một số điểm trong hệ thống.

2.3. Ước tính hiệu năng

Một trong những mục đích chính của mô phỏng là *ước tính* hiệu năng. Đối với hệ thống truyền thông tương tự, phép đo hiệu năng cơ bản là tỷ số tín hiệu trên tạp âm đầu ra SNR_0. Với hệ thống truyền thông số, phép đo hiệu năng là tỉ số lỗi bit BER hoặc tỉ số lỗi khung FER. Tỉ số tín hiệu trên tạp âm SNR cũng là phép đo hiệu năng thứ cấp trong các hệ thống truyền thông số (vì BER là một hàm đơn trị của SNR). Dùng kỹ thuật Monte Carlo để ước tính phép đo hiệu năng. Để minh họa các mặt của phương pháp luận cho mô phỏng Monte Carlo và ước tính hiệu năng, ta xét bài toán ước tính xác suất lỗi trong các hệ thống truyền thông số. Mô hình mô phỏng được cho ở hình 2.2. Lưu ý rằng, mô hình mô phỏng khi này là mô hình đơn giản của hệ thống được cho ở hình 2.1. Một số khối như đồng bộ và mã hoá được loại bỏ. Đồng bộ được coi là lý tưởng hoặc những ảnh hưởng của đồng bộ được xử lý thông qua việc điều kiện hóa và phân chia hóa như đã được giải thích trong phần trước. Mã hoá cũng được loại bỏ vì trọng tâm là tính toán xác suất lỗi không mã hoá trong hệ thống; các ảnh hưởng của việc mã hoá được xử lý riêng biệt trong chương 8. Ngoài ra, kênh được giả định là thay đổi *chậm* hoặc *tựa tĩnh*, và các trọng số bộ cân bằng là "*ổn định*" sau khi chúng hội tụ tới các giá trị có trạng thái ổn định.

Hiệu năng BER được mô phỏng bằng việc đưa chuỗi bit ngẫu nhiên vào bộ điều chế, qua các khối chức năng của mô hình được cho ở hình 2.2, sau đó được so sánh và đếm lỗi để xác định BER. Lưu ý rằng vì mục đích mô phỏng xác định BER không mã hóa kiểm soát lỗi nên không cần đến nguồn dữ liệu thực tế, bộ giải mã hoá nguồn, bộ mã hoá kiểm soát lỗi, bộ đan xen (vì ảnh hưởng của những khối này là tạo ra chuỗi nhị phân ngẫu nhiên). Việc đơn giản hóa này phải được thực hiện trước khi mô phỏng ước tính hiệu năng. Động cơ chủ yếu để đơn giản hoá là giảm thời gian mô phỏng. Vì vậy, chỉ các thành phần có ảnh hưởng đáng kể lên hiệu năng được chứa trong sơ đồ khối.



Hình 2.2: Mô hình mô phỏng để xác định BER

BER được xác định theo phương pháp Monte Carlo, BER không thể là tất định mà đúng hơn là được ước tính bằng cách cho N ký hiệu qua hệ thống và đếm lỗi. Giả sử rằng đếm được N_e lỗi trong số N ký hiệu đi qua hệ thống thì BER là:

$$\hat{P}_E = \frac{N_e}{N} \tag{2.6}$$

Là một ước tính của xác suất lỗi.

$$P_E = \lim_{N \to \infty} \frac{N_e}{N} \tag{2.7}$$

Tổng quát, ước tính Monte Carlo không bị lệch. Khi các giá trị của N nhỏ dẫn đến các ước tính lỗi có phương sai lớn (khi phương sai của ước tính lớn thì ước tính kém chính xác), và khi các giá trị của N là lớn cho ta các ước tính lỗi có phương sai nhỏ. Ước tính $\hat{P}_E$ hội tụ về P_E (giá trị đúng của xác suất lỗi) khi $N \to \infty$, vì vậy thực tế thường dùng N lớn nhất có thể (cần sự dung hòa các tham số đối nghịch: thời gian mô phỏng và tính chính xác của kết quả mô phỏng). Chương sau sẽ đề cập các kỹ thuật làm giảm phương sai của ước tính lỗi cho giá trị N cố định. Các kỹ thuật giảm phương sai, cần có sự kết hợp giữa phân tích và mô phỏng, và phải được áp dụng cẩn thận. Theo đó, cần có kiến thức khá sâu về lý thuyết ước tính.

Cần đặc biệt lưu ý hai khối chức năng ở hình 2.2 là những khối "*trễ*" và "*so sánh và đếm lỗi*". Đối với: *(i)* Khối "so sánh và đếm lỗi" có chức năng rõ ràng là so sánh các ký hiệu thu với các ký hiệu dữ liệu gốc sau đó đếm số lỗi N_e; *(ii)* Khối *"trễ"* xác định thời gian trễ của toàn bộ các phần tử của hệ thống để việc so sánh đếm lỗi được chính xác. Cần lưu ý rằng việc xác định trễ là khó và phải được thực hiện cẩn thận (bởi lẽ tính ngẫu nhiên của các phần tử trong hệ thống). Có thể được giải thích sơ bộ như sau: Một số khối chức năng có đáp ứng pha khác không, vì vậy tín hiệu đi qua những khối chức năng này bị trễ thời gian. Kết quả là tín hiệu tại đầu ra của khối nguồn dữ liệu phải được trễ sao cho ký hiệu tại đầu ra của máy thu so sánh được với ký hiệu tương ứng tại đầu ra của nguồn dữ liệu. Việc xác định lượng trễ này phải được thực hiện cẩn thận. Nếu không chính xác thì ước tính BER sẽ bị lệch, ước tính BER sẽ vượt quá xác suất lỗi đúng. Việc xác định giá trị trễ thích hợp là phần quan trọng của quy trình hiệu

chuẩn. Hiệu chuẩn mô phỏng là một thủ tục để đảm bảo các mức tín hiệu, các mức tạp âm, các độ trễ và các thuộc tính quan trọng khác trong mô phỏng hệ thống phù hợp với các thuộc tính tương ứng của hệ thống đang được mô phỏng. Chi tiết hóa cho bài toán này được đề cập ở chương 10, ở đó sẽ bàn về kỹ thuật mô phỏng Monte Carlo sâu hơn.

Khối trễ được thực thi là đường trễ chiều dài khả biến. Chiều dài đối với một ứng dụng cụ thể được chọn để nhận được sự đồng chỉnh phù hợp giữa ký hiệu được giải điều chế (tại đầu ra máy thu) với ký hiệu phát (tại đầu ra nguồn dữ liệu). Độ trễ thường được lượng tử thành một số nguyên lần chu kỳ lấy mẫu. Để điều khiển tốt về độ trễ cần có chu kỳ lấy mẫu rất ngắn, (tần số lấy mẫu rất lớn) trong mô phỏng. Việc tăng tần số lấy mẫu làm tăng thời gian mô phỏng. Cần phải nghiên cứu bài toán tối thiểu thời gian chạy mô phỏng nhưng vẫn đảm bảo độ chính xác, đây là vấn đề có *tính nghệ thuật* của mô phỏng.

Chương 3
LẤY MẪU VÀ LƯỢNG TỬ

3.1. Mở đầu

Mục đích chính của cuốn sách là nghiên cứu các kỹ thuật cơ bản để mô phỏng hệ thống truyền thông trên máy tính điện tử số ở mức chính xác nhất có thể. Hầu hết các ứng dụng, đều phải tạo và xử lý dạng sóng. Máy tính chỉ có thể xử lý các con số (biểu diễn các mẫu tín hiệu). Ngoài ra, do tính hữu hạn về độ dài từ mã của máy tính, nên các giá trị mẫu có độ chính xác hữu hạn. Nói cách khác, các giá trị của mẫu được lượng tử hóa. Do vậy, lấy mẫu và lượng tử hóa là các hoạt động căn bản trong quá trình mô phỏng, và chúng gây thêm sai số vào kết quả mô phỏng. Việc khử hoàn toàn các sai số này là không thể, và cần phải có giải pháp dung hòa. Có thể thấy rằng: cách tốt nhất là giảm thiểu các ảnh hưởng gây sai số của quá trình lấy mẫu và lượng tử. Cũng cần lưu ý rằng, các hệ thống vật lý sử dụng kỹ thuật xử lý tín hiệu số DSP cũng phải chịu các ảnh hưởng của sai số lượng tử và lấy mẫu.

Vì vậy, chương này xét một số chủ đề lấy mẫu và trình bày các giá trị mẫu trong mô phỏng hệ thống truyền thông. Xét vắn tắt hai định lý lấy mẫu cơ bản là định lý lấy mẫu thông thấp và định lý lấy mẫu thông dải. Vì các tín hiệu thông dải thường được biểu diễn bởi tín hiệu thông thấp trong các mô phỏng hệ thống, nên định lý lấy mẫu thông thấp là định lý quan trọng nhất. Ta sẽ thấy rằng, có thể lấy mẫu tín hiệu thông thấp băng tần hạn chế, và có thể khôi phục tín hiệu thông dải từ các giá trị mẫu nếu tần số lấy mẫu lớn hơn 2 lần độ rộng băng tần của tín hiệu thông thấp hạn băng. Định lý lấy mẫu thông dải, mặc dù không hữu dụng lắm so với định lý thông thấp trong mô phỏng, nhưng mang lại kết quả tương đương. Tín hiệu thông dải có thể được lấy mẫu và được tái tạo nếu tần số lấy mẫu nằm trong khoảng 2B và 4B trong đó B là độ rộng băng tần của tín hiệu thông dải được lấy mẫu.

Tiếp theo là xét quá trình lượng tử. Sai số lượng tử xuất hiện trong mọi mô phỏng, vì giá trị mẫu chỉ được biểu diễn bởi các từ mã chiều dài hữu hạn. Xét hai loại sai số lượng tử: sai số do biểu diễn số với dấu phẩy tĩnh và sai số lượng tử do biểu diễn số bằng dấu phẩy động. Ta sẽ thấy rằng, khi sử dụng biểu diễn số dấu phẩy tĩnh, thì tỉ số tín hiệu trên tạp âm lượng tử tăng 6 dB mỗi khi chiều dài từ mã tăng thêm một bit. Nói chung, khi mô phỏng trên các máy tính sử dụng biểu diễn dấu phẩy động, thì thường bỏ qua tạp âm lượng tử được. Tuy nhiên, tạp âm này không bao giờ bằng 0, tích lũy dần và làm giảm đáng kể độ chính xác kết quả mô phỏng. Ta nên cảnh giác với nguồn nhiễu này.

Sau đó là đề cập vấn đề *tái tạo* và *nội suy*. Ta sẽ thấy rằng, nếu lấy mẫu tín hiệu thông thấp hạn băng với tần số lấy mẫu lớn hơn 2 lần độ rộng băng thông của tín hiệu đó, thì có thể khôi phục (tái tạo) tín hiệu liên tục cơ bản đó mà không có sai số bằng cách đánh trọng số cho mỗi mẫu với một sóng dạng $\sin(x)/x$, tương đương với việc cho các mẫu tín hiệu đó qua bộ lọc thông thấp lý tưởng. Kết quả là dạng sóng được xác định cho tất cả các giá trị theo thời gian và bằng cách tách các mẫu *"mới"* giữa các mẫu gốc, ta có thể tạo các mẫu nội suy. Quá trình

này gọi là tăng mẫu, làm tăng tần số lấy mẫu hiệu quả. Quá trình ngược lại là giảm mẫu thực hiện bằng cách trích (tách) mỗi mẫu thứ M từ một tập các mẫu gốc. Triển khai sử dụng tăng mẫu (upsampling) và giảm mẫu (downsampling) trong mô phỏng dẫn đến xuất hiện nhiều tần số lấy mẫu, chúng hữu hiệu khi mô phỏng hệ thống các tín hiệu băng tần khác nhau lớn ví dụ như hệ thống thông tin trải phổ.

Chủ đề quan trọng là quan hệ giữa tần số lấy mẫu với dạng xung sử dụng để truyền dẫn dạng sóng. Dạng xung được giả thiết là giới hạn về thời gian, và do vậy nó không thể được giới hạn băng (*băng tần vô hạn*). Vì vậy, xuất hiện các sai số chồng phổ. Tiêu chuẩn để lựa chọn tần số lấy mẫu là xác định tỉ số tín hiệu trên tạp âm cần thiết, ở đó nguồn tạp âm gây ra sai số chồng phổ (sai số chồng phổ tạo thành nguồn tạp âm). Phương pháp chung để xác định PSD của tín hiệu điều chế và tích phân số của PSD này xác định tỉ số tín hiệu trên tạp âm chồng phổ.

3.2. Lấy mẫu

Quá trình biến đổi tín hiệu tương tự thành tín hiệu số theo trình tự: lấy mẫu, lượng tử hóa và mã hóa được minh họa ở hình 3.1. Tín hiệu tương tự *x(t)* liên tục cả về thời gian và độ lớn. Kết quả của quá trình lấy mẫu là các tín hiệu rời rạc về thời gian nhưng liên tục về biên độ được gọi là tín hiệu số liệu mẫu. Tín hiệu số được tạo ra từ tín hiệu dữ liệu mẫu bằng cách mã hóa các giá trị mẫu thành tập các giá trị hữu hạn. Ta sẽ thấy, các lỗi sẽ xảy ra tại mỗi bước của quá trình này.

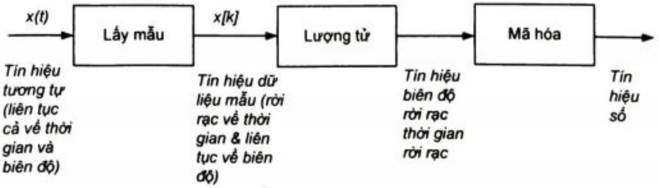
3.2.1. Định lý lấy mẫu thông thấp

Bước đầu tiên trong quá trình tạo tín hiệu số từ tín hiệu liên tục *x(t)* là lấy mẫu *x(t)* tại các điểm cách đều theo thời gian để tạo ra các giá trị mẫu $x_s(t) = x(kT_s) = x[k]$. Tham số T_s là chu kỳ lấy mẫu, là nghịch đảo của tần số lấy mẫu f_s.

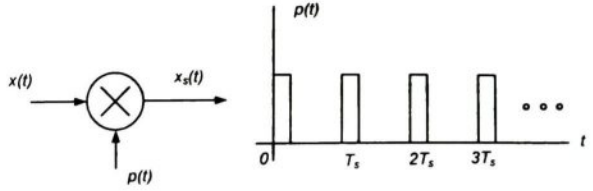
Mô hình lấy mẫu được minh họa ở hình 3.2. Tín hiệu *x(t)* được nhân với một xung tuần hoàn *p(t)* tạo ra tín hiệu mẫu $x_s(t)$. Nói cách khác:

$$x_s(t) = x(t)p(t) \tag{3.1}$$

Hàm lấy mẫu *p(t)* được giả thiết là một xung hẹp nhận giá trị 1 hoặc 0. Theo đó, $x_s(t) = x(t)$ khi $p(t) = 1$ và $x_S(t) = 0$ khi $p(t) = 0$. Cần đặc biệt lưu ý đến chu kỳ của hàm lấy mẫu *p(t)* còn dạng sóng của *p(t)* là tùy chọn. Hàm lấy mẫu được minh họa trong hình 3.2, trực giác, nó thể hiện một cái chuyển mạch tại các thời điểm lấy mẫu.



Hình 3.1: Lấy mẫu, lượng tử, mã hóa



a) Hoạt động lấy mẫu b) Hàm lấy mẫu

Hình 3.2: Lấy mẫu và hàm lấy mẫu

Vì $p(t)$ là hàm tuần hoàn, nên được biểu diễn bằng chuỗi Fourier:

$$p(t) = \sum_{n=-\infty}^{\infty} C_n . e^{j2\pi n f_s t} \tag{3.2}$$

Trong đó, các hệ số chuỗi Fourier được cho bởi:

$$C_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} p(t).e^{-j2\pi n f_s t}\, dt \tag{3.3}$$

Thay (3.2) vào (3.1) ta có:

$$x_s(t) = x(t) \sum_{n=-\infty}^{\infty} C_n e^{j2\pi n f_s t} \tag{3.4}$$

Để rút ra định lý lấy mẫu và các điều kiện để khôi phục tín hiệu $x(t)$ từ các mẫu $x(kT_s)$, cần phải xác định phổ của $x_s(t)$ và chỉ ra rằng $x(t)$ có thể khôi phục từ $x_s(t)$ với các điều kiện cần thiết. Muốn vậy, cần phải xác định: *(i)* phổ của tín hiệu mẫu $x_s(t)$; *(ii)* phổ của tín hiệu thông tin $x(t)$; *(iii)* phổ của xung lấy mẫu $p(t)$; *(iv)* các mối quan hệ từ đó tìm được điều kiện lấy mẫu hay định lý lấy mẫu. Theo đó, ta tiến hành như sau:

Phổ của tín hiệu $x_s(t)$: Theo định nghĩa, biến đổi Fourier của tín hiệu mẫu là:

$$X_s(f) = \int_{-\infty}^{\infty} \underbrace{x(t) \sum_{n=-\infty}^{\infty} C_n e^{j2\pi n f_s t}}_{x_s(t)} e^{-j2\pi f t}\, dt \tag{3.5}$$

Chuyển đổi thứ thự lấy tích phân và lấy tổng ta được:

$$X_s(f) = \sum_{n=-\infty}^{\infty} C_n \int_{-\infty}^{\infty} x(t) e^{-j2\pi(f - n f_s)t}\, dt \tag{3.6}$$

Phổ của tín hiệu tương tự $x(t)$: Phổ của $x(t)$ được xác định theo định nghĩa biến đổi Fourier là:

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t}\, dt \tag{3.7}$$

Từ (3.6), ta có biến đổi Fourier của tín hiệu mẫu được viết như sau:

$$X_s(f) = \sum_{n=-\infty}^{\infty} C_n X(f - nf_s) \qquad (3.8)$$

Từ (3.8), thấy rõ quan hệ giữa phổ của tín hiệu mẫu $x_s(t)$ và phổ của tín hiệu tương tự $x(t)$. Theo đó, ảnh hưởng của việc lấy mẫu tín hiệu liên tục là tái sinh phổ của tín hiệu được lấy mẫu xung quanh tần số 0 ($f = 0$) và tất cả các hài của tần số lấy mẫu ($f = nf_s$). Phổ được dịch, được đánh trọng số bởi hệ số chuỗi Fourier C_n của xung lấy mẫu $p(t)$ tương ứng.

Phổ của xung lấy mẫu p(t): Vì các mẫu dữ liệu phải có giá trị bằng với giá trị của tín hiệu được lấy mẫu tại các thời điểm lấy mẫu *và* phải có chu kỳ lặp lại bằng với chu kỳ của xung lấy mẫu, nên định phù hợp cho xung lấy mẫu $p(t)$ là:

$$p(t) = \sum_{k=-\infty}^{\infty} \delta(t-kT_s) \qquad (3.9)$$

được coi là lấy mẫu hàm xung kim trong đó các giá trị mẫu được thể hiện bởi các trọng số của các hàm xung kim. Thế (3.9) vào (3.3), nhận được:

$$C_n = \frac{1}{T_s} \int_{-T_s/2}^{T_s/2} \delta(t) e^{-j2\pi n f_s t}\, dt \qquad (3.10)$$

Áp dụng tính chất dịch của hàm delta, ta có:

$$C_n = \frac{1}{T_s} = f_s \qquad (3.11)$$

Sử dụng kết quả này vào (3.2) và biến đổi Fourier cho $p(t)$ nhận được:

$$P(f) = f_s \sum_{n=-\infty}^{\infty} \delta(f - nf_s) \qquad (3.12)$$

Rút ra điều kiện lấy mẫu: Vì lấy mẫu hàm xung kim $C_n = f_s$, $\forall n$. Nên phổ của tín hiệu mẫu $X_s(f)$ theo (3.8) trở thành:

$$X_s(f) = f_s \sum_{n=-\infty}^{\infty} X(f - nf_s) \qquad (3.13)$$

Lưu ý, có thể tìm được phổ tín hiệu mẫu $X_s(f)$ theo biểu thức sau:

$$X_s(f) = X(f) \otimes P(f) \qquad (3.14)$$

trong đó $\otimes$ ký hiệu cho tích chập, tạo $X_s(f)$ theo (3.14) được cho ở hình 3.3 đối với trường hợp tín hiệu giới hạn băng.

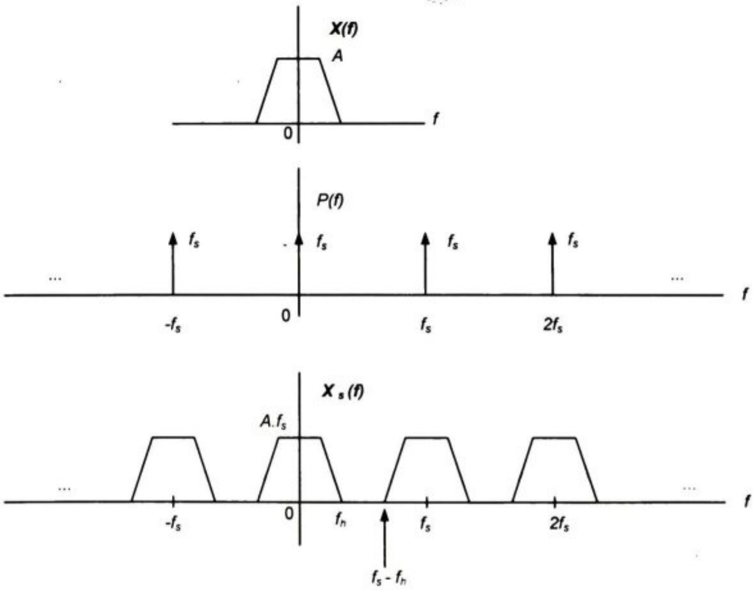
Định lý lấy mẫu có thể được triển khai từ việc quan sát hình 3.3. Để các mẫu $x(nT_s)$ chứa toàn bộ thông tin của tín hiệu $x(t)$ (quá trình lấy mẫu không làm mất thông tin) thì lấy mẫu phải được thực hiện sao cho có thể khôi phục $x(t)$ từ các mẫu $x(nT_s)$ mà không có lỗi. Việc tái tạo

x(t) từ *x(nT$_s$)* được thực hiện bằng cách tách thành phần $n = 0$ từ $X_s(f)$ bằng cách lọc thông thấp. Để khôi phục tín hiệu *x(t)* từ *x(nT$_s$)* mà không có lỗi, thì vùng phổ của $X_s(f)$ được định vị xung quanh $f = \pm f_s$ [thành phần $n = \pm 1$ trong (3.13)] không chồng lấn lên vùng phổ của $X_s(f)$ được định vị xung quanh $f = 0$ [thành phần n=0 trong (3.13)], nói cách khác, tất cả các thành phần phổ được dịch trong (3.13) phải không chồng lấn nhau. Vì vậy cần phải đảm bảo điều kiện $f_s - f_h > f_h$ hay $f_s > 2f_h$.

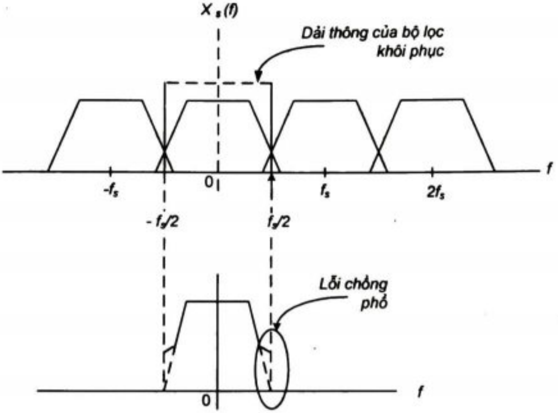
Định lý 3.1: *Có thể khôi phục tín hiệu băng tần hạn chế từ các mẫu của tín hiệu đó mà không có lỗi nếu tần số lấy mẫu f_s lớn hơn $2f_h$, trong đó f_h là tần số lớn nhất trong tín hiệu được lấy mẫu.*

Định lý 3.1 thường được gọi là định lý lấy mẫu thông thấp, nó cũng đúng đối với tín hiệu thông dải. Tuy nhiên, khi sử dụng định lý lấy mẫu thông thấp cho tín hiệu *thông dải* dẫn đến tần số lấy mẫu rất cao. Lấy mẫu tín hiệu thông dải sẽ được đề cập dưới đây.

Nếu $f_s < 2f_h$ thì phần phổ có trung tâm tại $f = \pm f_s$ sẽ chồng lấn lên phần phổ có trung tâm tại $f = 0$ dẫn đến hậu quả là gây méo tín hiệu tại đầu ra *x(t)* của bộ lọc thông thấp như minh họa trong hình 3.4 (giả sử phổ *x(t)* là thực). Méo dạng này gọi là chồng phổ.



Hình 3.3: Lấy mẫu trong miền tần số



Hình 3.4: Tần số lấy mẫu thấp dẫn đến lỗi chồng phổ

3.2.2. Lấy mẫu tín hiệu ngẫu nhiên thông thấp

Dạng sóng $x(t)$ ở trên được giả thiết là *tín hiệu năng lượng hữu hạn tất định*. Kết quả của các giả định này là, tồn tại biến đổi Fourier và định lý lấy mẫu dựa vào phổ (biến đổi Fourier) của tín hiệu. Trong hầu hết các ứng dụng, thường phải mô phỏng *xử lý* các hàm mẫu của một quá trình ngẫu nhiên. Vì vậy, thay vì chọn tần số lấy mẫu dựa vào phổ của tín hiệu tương tự (bằng cách biến đổi Fourier hàm tất định), ta chọn tần số lấy mẫu dựa vào mật độ phổ công suất PSD của tín hiệu ngẫu nhiên.

Trường hợp các tín hiệu *ngẫu nhiên* ta viết:

$$X_S(t) = X(t)P(t) \tag{3.15}$$

Hàm lấy mẫu $P(t)$ khi này được viết là:

$$P(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s - D) \tag{3.16}$$

Trong đó D là biến ngẫu nhiên phân bố đều trong khoảng $(0,T_s)$ và độc lập với $X(t)$. Lưu ý tính giống nhau giữa (3.15) và (3.1), giữa (3.16) và (3.9). Chỉ có 2 điểm khác biệt cơ bản là: *(i)* sử dụng các chữ in hoa trong các hàm thời gian $X(t)$, $P(t)$ và $X_S(t)$ để ký hiệu cho các quá trình ngẫu nhiên; *(ii)* sử dụng biến ngẫu nhiên D trong (3.16) nhằm đảm bảo cho $X_s(t)$ là quá trình ngẫu nhiên *dừng*. Nếu không có D, thì tín hiệu mẫu là quá trình dừng tuần hoàn (*cyclostationary*). Tác dụng của D là làm ngẫu nhiên gốc thời gian của $P(t)$.

Mật độ phổ công suất của $X_s(t)$ tìm được thông qua hàm tự tương quan của:

$$X_s(t) = X(t)\sum_{k=-\infty}^{\infty} \delta(t - kT_s - D) \tag{3.17}$$

Biến đổi Fourier hàm tự tương quan ta được mật độ phổ công suất PSD của $X_s(t)$ như sau:

$$S_{X_s}(f) = f_S^2 \sum_{n=-\infty}^{\infty} S_X(f - nf_S) \tag{3.18}$$

Trong đó $S_X(f)$ là PSD của $X(t)$. Cần phải lưu ý: *(i)* khía cạnh giống nhau của (3.18) và (3.13); *(ii)* hình 3.3 và 3.4 áp dụng nếu phổ là tương ứng với PSD của $X(t)$ và nếu các trục tọa độ được đánh nhãn phù hợp; *(iii)* vẫn dùng định lý lấy mẫu trước, và tín hiệu phải được lấy mẫu ở tần số lớn hơn gấp 2 lần tần số lớn nhất của tín hiệu để tránh méo do chồng phổ.

3.2.3. Lấy mẫu tín hiệu thông dải

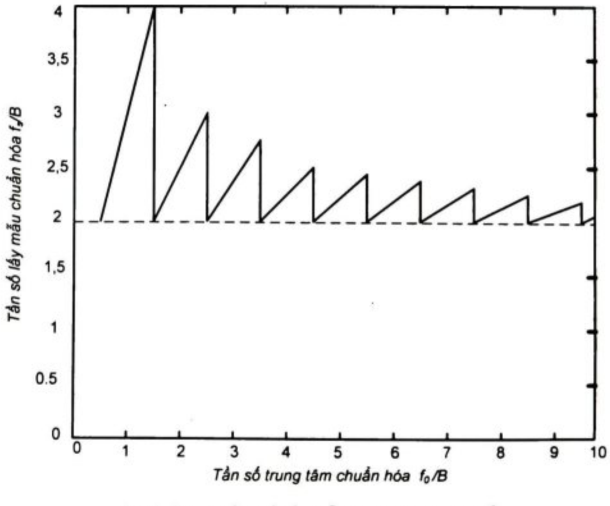
Xét lấy mẫu tín hiệu thông dải. Có nhiều cách để biểu diễn tín hiệu thông dải bằng một tập các mẫu. Dưới đây xét 2 phương pháp phổ biến nhất:

Định lý lấy mẫu thông dải:

Định lý lấy mẫu thông dải cho các tín hiệu thông dải giá trị thực được phát biểu như sau:

Định lý 3.2: *Nếu một tín hiệu thông dải có độ rộng băng tần là B và có tần số cao nhất là f_h, thì tín hiệu đó có thể được lấy mẫu và được khôi phục sử dụng tần số lấy mẫu $f_s = 2f_h/m$, trong đó m là số nguyên lớn nhất nhỏ hơn f_h/B. Tất cả các tần số lấy mẫu cao hơn không nhất thiết sử dụng trừ khi chúng vượt quá $2f_h$, là giá trị của f_s được lấy theo định lý lấy mẫu thông thấp.*

Tần số lấy mẫu chuẩn hóa f_s/B là hàm của tần số trung tâm chuẩn hóa f_0/B được minh họa ở hình 3.5, trong đó quan hệ giữa f_0 và f_h là $f_h = f_0+B/2$. Ta thấy rằng, tần số lấy mẫu khả dụng nằm trong khoảng $2B \le f_s \le 4B$. Tuy nhiên, trường hợp điển hình khi $f_0 >> B$, thì tần số lấy mẫu được xác định theo định lý lấy mẫu thông dải là xấp xỉ bằng, nhưng được giới hạn dưới là $2B$.



Hình 3.5: Tần số lấy mẫu tín hiệu thông dải

Lấy mẫu các tín hiệu đồng pha/vuông pha

Giả sử ta có một tín hiệu thông dải được biểu diễn ở dạng:

$$x(t) = A(t)\cos\left[2\pi f_c t + \phi(t)\right] \tag{3.19}$$

Các hàm $A(t)$ và $\phi(t)$ là đường bao và độ lệch pha của tín hiệu thông dải. Trong hầu hết các ứng dụng truyền thông cả $A(t)$ và $\phi(t)$ là tín hiệu *thông thấp* và có băng thông xấp xỉ băng thông của tín hiệu mang tin. Sử dụng các tính chất lượng giác, tín hiệu thông dải được viết là:

$$x(t) = \underbrace{A(t)\cos\phi(t)}_{x_d(t)}\cos 2\pi f_c t - \underbrace{A(t)\sin\phi(t)}_{x_q(t)}\sin 2\pi f_c t \tag{3.20}$$

$$x(t) = x_d(t)\cos 2\pi f_c t - x_q(t)\sin 2\pi f_c t \tag{3.21}$$

Theo đó, các thành phần đồng pha $x_d(t)$ và vuông pha $x_q(t)$ được định nghĩa như sau:

$$x_d(t) = A(t)\cos\phi(t) \tag{3.22}$$

$$x_q(t) = A(t)\sin\phi(t) \tag{3.23}$$

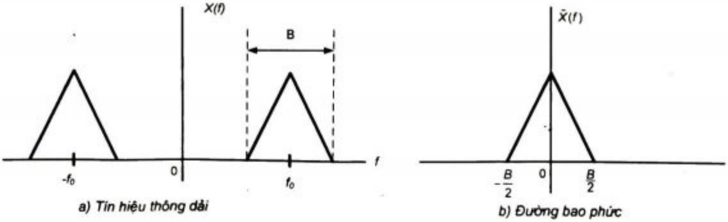
Vì $A(t)$ và $\phi(t)$ đều là tín hiệu thông thấp, nên $x_d(t)$ và $x_q(t)$ cũng là các tín hiệu thông thấp, vì vậy chúng phải được lấy mẫu theo định lý lấy mẫu thông thấp. Lưu ý rằng nếu biết trước $x_d(t), x_q(t)$ và tần số sóng mang f_c, thì tín hiệu thông dải (3.21) có thể được khôi phục mà không có lỗi. Trình bày các tín hiệu *thông dải* theo các thành phần đồng pha và vuông pha sẽ được khảo sát chi tiết ở chương 4.

Biểu diễn tín hiệu thông dải trong miền tần số được cho ở hình 3.6(a), đường bao phức tương ứng được định nghĩa bởi:

$$\tilde{x}(t) = x_d(t) + jx_q(t) \tag{3.24}$$

Vì cả $x_d(t)$ và $x_q(t)$ đều là tín hiệu thông thấp, nên:

$$\tilde{X}(f) = X_d(f) + jX_q(t) \tag{3.25}$$



Hình 3.6: Tín hiệu thông dải và đường bao phức

Cũng là tín hiệu thông thấp như được minh họa ở hình 3.6(b). Trong hình 3.6 ta thấy rằng $\tilde{X}(t)$, $x_d(t)$ và $x_q(t)$ đều là tín hiệu thông thấp. Vì vậy, $x_d(t)$ và $x_q(t)$ phải được lấy mẫu theo định lý lấy mẫu thông thấp. Vì tần số cao nhất trong $x_d(t)$ và $x_q(t)$ là $B/2$, nên tần số lấy mẫu nhỏ nhất cho mỗi thành phần này là B. Tuy nhiên, cả hai tín hiệu thông thấp $x_d(t)$ và $x_q(t)$ đều phải được lấy mẫu. Kết quả là, phải dùng tốc độ lấy mẫu lớn hớn 2B. Vì vậy, việc lấy mẫu đường bao phức theo *định lý lấy mẫu thông thấp*, mang lại sử dụng tần số lấy mẫu đó để lấy mẫu tín hiệu thông dải giá trị thực theo theo định lý lấy mẫu thông dải cho trường hợp $f_0 >> B$.

Ví dụ 3.1: Thấy rõ, có thể khôi phục tín hiệu thông dải $x(t)$ mà không có lỗi nếu $x_d(t)$ và $x_q(t)$ được lấy mẫu theo định lý lấy mẫu thông thấp một cách thích hợp. Ưu điểm khi biểu diễn tín hiệu thông dải bằng thông thấp tương đương là rõ ràng. Ta biểu diễn tín hiệu FM bởi một tập các mẫu trong một giây. Giả sử, tần số sóng mang là 100 MHz (băng tần phát thanh FM) tần số cao nhất trong tín hiệu mang tin là 15 kHz. Băng thông B của tín hiệu điều chế thường được lấy xấp xỉ theo nguyên tắc của Carson đó là:

$$B = 2(D+1)W \tag{3.26}$$

Giả sử tỉ lệ độ lệch $D = 5$ nên ta có:

$$B = 2(5+1)15 = 180\ kHz \tag{3.27}$$

Theo đó, tần số cao nhất trong tín hiệu điều chế là $f_c + B/2 = 100.000 + 90 = 100.090$ kHz. Vậy, theo định lý lấy mẫu thông thấp thì trong một giây tín hiệu phải có tối thiểu là $2 * (f_c + B/2) = 200.180.000$ mẫu.

Ta giả sử, biểu diễn tín hiệu FM theo thành phần đồng pha và vuông pha. Băng thông của $x_d(t)$ và $x_q(t)$ đều là $B/2 = 90$ kHz. Do vậy mỗi tín hiệu $x_d(t)$ và $x_q(t)$ cần ít nhất 180.000 mẫu để biểu diễn một giây tín hiệu. Điều này mang lại là 360.000 mẫu thông thấp để biểu diễn một giây dữ liệu. Nó sẽ tiết kiệm được:

$$\frac{200,180,000}{360,000} \approx 556 \tag{3.28}$$

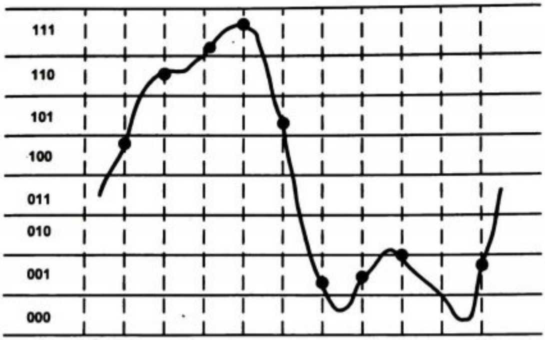
Do đó, giảm thời gian chạy của máy tính.

3.3. Lượng tử

Quá trình lượng tử và mã hóa được minh họa trong hình 3.7, cho thấy dạng sóng liên tục, tín hiệu mẫu được thể hiện bằng các dấu chấm đen, mỗi mẫu tín hiệu nằm trong một mức lượng tử cụ thể. Nếu lượng tử n mức và mỗi mức lượng tử được biểu diễn bởi một từ mã nhị phân b bit, thì ta có quan hệ

$$n = 2^b \tag{3.29}$$

Trong hình 3.7, mỗi một mức lượng tử được gán bằng một từ mã 3 bit ($b = 3$ và $n = 8$). Sau khi lượng tử hóa, các giá trị mẫu được biểu diễn bởi các từ mã ứng với mức lượng tử đó và số hóa dạng sóng tín hiệu bằng các từ mã này. Ví dụ, ba giá trị mẫu đầu tiên từ trái sang phai trong hình 3.7 được biểu diễn bằng chuỗi nhị phân $\underbrace{100}\underbrace{110}\underbrace{111}$.

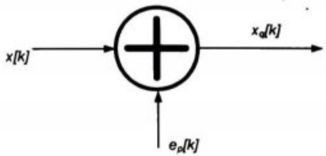


Hình 3.7: Lượng tử và mã hóa

Từ lý thuyết lấy mẫu, nếu lấy mẫu các tín hiệu tương tự băng thông hạn chế tại tần số lớn hơn tần số Nyquist thì khôi phục tín hiệu từ các mẫu mà không có lỗi. Vì vậy dưới các điều kiện này, hoạt động lấy mẫu là khả nghịch. Tuy nhiên, hoạt động lượng tử là không khả nghịch. Một khi các giá trị mẫu được lượng tử, chỉ có mức lượng tử được duy trì, và do vậy gây ra một lỗi ngẫu nhiên. Nếu ký hiệu $x[k]$ là giá trị tín hiệu tại thời điểm lấy mẫu $t = kT_s$, $x_q[k]$ là giá trị được lượng tử hóa, $e_q[k]$ là lỗi lượng tử, thì:

$$x_q[k] = x[k] + e_q[k] \tag{3.30}$$

Mô hình bộ lượng tử hóa cho (3.30) được minh họa ở hình 3.8. Nếu tín hiệu gốc không được hạn băng, thì tín hiệu số chứa cả lỗi lượng tử và lỗi chồng phổ.



Hình 3.8: Mô hình hóa lỗi lượng tử

Khi xét tỉ số tín hiệu trên tạp âm SNR, tạp âm được hiểu là do quá trình lượng tử hóa gây ra. Nếu ký hiệu $E\{\}$ là kỳ vọng thống kê, N_q là công suất tạp âm do lượng tử hóa, thì tỉ số tín hiệu trên tạp âm do lượng tử hóa $(SNR)_q$ được định nghĩa bởi:

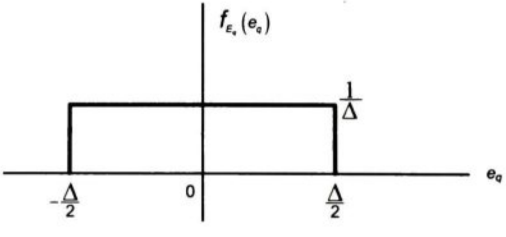
$$(SNR)_q = \frac{S}{N_q} = \frac{E\left\{x^2[k]\right\}}{E\left\{e_q^2[k]\right\}} \tag{3.31}$$

Theo đó, để xác định $(SNR)_q$, cần phải biết hàm mật độ xác suất của lỗi $e_q[k]$. Hàm mật độ xác suất pdf của lỗi lượng tử là hàm của khuôn dạng được dùng để biểu diễn các con số trong máy tính. Có nhiều khuôn dạng được sử dụng. Loại phổ biến nhất là loại dấu phẩy động và dấu phẩy tĩnh.

Số học dấu phẩy tĩnh

Khi dùng máy tính số để mô phỏng thì các con số thường được trình bày ở dạng dấu phẩy động nhưng có nhiều lý do phải trình bày con số ở dạng dấu phẩy tĩnh (lỗi lượng tử). Một số lý giải cho việc dùng số học dấu phẩy tĩnh: *(i)* Minh họa các cơ chế cơ bản gây lỗi lượng tử hóa, các phép tính dấu phẩy tĩnh chạy nhanh hơn rất nhiều so với các phép tính dấu phẩy động; *(ii)* Tiêu thụ công suất của các vi xử lý dùng dấu phẩy tĩnh nhỏ hơn; *(iii)* Khi mô phỏng các thiết bị sử dụng dấu phẩy tĩnh. Chẳng hạn, các hệ thống truyền thông dựa trên phần mềm đang trở nên phổ biến (bởi lẽ tính khả lập lại cấu hình tương thích). Giá thành thiết bị được giảm thiểu khi dùng tính toán dấu phẩy tĩnh, hơn nữa các phép toán dấu phẩy tĩnh chạy nhanh hơn rất nhiều so với các phép toán dấu phẩy động. Cần lưu ý rằng, việc thiết kế các thiết bị dựa trên phần mềm thường bắt đầu bằng việc mô phỏng, nếu kết quả mô phỏng cho thấy thiết bị hoạt động đạt chỉ tiêu thiết kế, thì mã mô phỏng được nạp vào các thiết bị đó. Trong các ứng dụng như vậy, việc mô phỏng các thiết bị và thiết bị vật lý hợp nhất vào một quy mô lớn. Như đã đề cập, chỉ tiêu tốc độ, giá cả, tiêu thụ công suất dẫn đến thương mại hóa, hầu hết các thiết bị đều khai thác tính toán dấu phẩy tĩnh, mô phỏng là một công cụ quan trọng thiết kế và định lượng hiệu năng của thiết bị.

Nếu ký hiệu Δ là độ rộng của mức lượng tử (xem hình 3.7) và giả thiết giá trị của mẫu tương ứng với một mức lượng tử nhận giá trị trung tâm của mức lượng tử đó thì lỗi lượng tử lớn nhất $|e_q[k]|$ là $\Delta/2$. Nếu số mức lượng tử lớn thì từ mã tương ứng sẽ dài. Nếu tín hiệu thay đổi nhiều giữa các mẫu liền kề, thì khả năng một mẫu cho trước rơi vào bất kỳ một điểm nào đó trong mức lượng tử là như nhau. Khi này lỗi do lượng tử hóa được coi là phân bố đều và độc lập. Theo đó, hàm mật độ xác suất pdf của lỗi lượng tử là một hàm phân bố đều trong khoảng $[-\Delta/2, \Delta/2]$ được minh họa trong hình 3.9. Nếu ký hiệu lỗi lượng tử của mẫu thứ k là $e_q[k]$ thì ta có: trung bình của lỗi lượng tử; phương sai của lỗi lượng tử; $(SNR)_q$ như sau:



Hình 3.9: Hàm mật độ xác suất giả định của lỗi lượng tử

Trung bình của lỗi lượng tử $e_q[k]$ là:

$$E\{e_q[k]\} = \int_{-\Delta/2}^{\Delta/2} x\frac{1}{\Delta}dx = 0 \tag{3.32}$$

Lỗi lượng tử có trung bình 0.

Phương sai của lỗi lượng tử $e_q[k]$ là:

$$E\left(e_q^2[k]\right) = \int_{-\Delta/2}^{\Delta/2} x^2 \frac{1}{\Delta} dx = \frac{\Delta^2}{12} \tag{3.33}$$

Tính tỉ số tín hiệu trên tạp âm do lượng tử hóa

Nếu giả thiết và ký hiệu: *(i) D* là dải động bộ lượng tử hóa (dải động là giá trị ***đỉnh-đỉnh*** của tín hiệu vào bộ lượng tử hóa); *(ii) b* là chiều dài của từ mã; *(iii)* nếu dùng số học nhị phân, thì có 2^b mức lượng tử có thể có và dải động của nó được cho bởi:

$$D = 2^b \Delta \tag{3.34}$$

Do vậy:

$$\Delta = D2^{-b} \tag{3.35}$$

và công suất tạp âm do lượng tử là:

$$N_q = E\left(e_q^2[k]\right) = \frac{D^2}{12} 2^{-2b} \tag{3.36}$$

Nếu ký hiệu *S* là công suất tín hiệu và $(SNR)_q$ là tỉ số tín hiệu trên tạp âm do lượng tử hóa, thì ta có:

$$(SNR)_q = \frac{S}{\left(\frac{D^2}{12}\right) 2^{-2b}} = \frac{12S}{D^2} 2^{2b} \tag{3.37}$$

Giả sử tín hiệu có trung bình 0, quan hệ giữa giá trị *S* và *D* được cho bởi hệ số gợn sóng (*Crest Factor*) của tín hiệu cơ bản. Hệ số gợn sóng được định nghĩa là tỉ số của RMS (hay độ lệch chuẩn) của tín hiệu *với* giá trị đỉnh của tín hiệu đó. Để minh họa quan hệ này, giả sử tín hiệu cơ bản có dải rộng *D*, nằm trong dải ± *D*. Vì công suất tín hiệu này là *S*, độ lệch chuẩn là $\sqrt{S}$, do vậy hệ số gợn sóng là:

$$F_c = \frac{\sqrt{S}}{D/2} = \frac{2\sqrt{S}}{D} \tag{3.38}$$

Thay (3.38) vào (3.37) ta có:

$$(SNR)_q = 3F_c^2 2^{2b} \tag{3.39}$$

Chuyển sang đơn vị dB là:

$$(SNR)_q = 4,7712 + 20\log_{10} F_c + 6,0206b, \ dB \tag{3.40}$$

Lưu ý rằng, các tín hiệu có hệ số gợn sóng lớn thì dễ tránh lỗi lượng tử hơn. Điều này là hợp logic bởi lẽ các tín hiệu có hệ số gợn sóng lớn có độ lệch chuẩn lớn, nghĩa là chúng được

trải rộng trên các mức lượng tử hơn. Tuy vậy, chiều dài từ mã **b** có ảnh hưởng lên $(SNR)_q$ nhất, tăng bit thêm vào chiều dài từ mã thì $(SNR)_q$ tăng 6 dB.

Số học dấu phẩy động

Khi mô phỏng trên máy tính sử dụng dấu phẩy động, khuôn dạng của các con số dấu phẩy động là $\pm M*(\pm 10\wedge E)$, trong đó M là phần định trị và E là số mũ. Yêu cầu về độ chính xác, sử dụng các từ số 64-bit, 64 bit phải được phân bổ giữa phần định trị *(mantissa)* và số mũ. Việc phân bổ này có ảnh hưởng lớn đến kết quả tính toán. Tuy nhiên, việc ấn định này hầu như được tiêu chuẩn hóa, nhưng không phải tất cả. Tiêu chuẩn ANSI/IEEE đối với đại số dấu phẩy động định rõ 53 bit định trị và 11 bit số mũ. Matlab cung cấp một phương pháp đơn giản để xác định dù tiêu chuẩn IEEE có được thực hiện trên máy tính hay không bằng lệnh *isieee*.

Vì sẽ dùng Matlab để triển khai và giải thích các mô phỏng nên cần phải xét độ chính xác. Theo đó, tham số quan trọng nhất từ việc sử dụng dấu phẩy động là độ phân giải (sai khác giữa 1 và con số dấu phẩy động lớn nhất tiếp theo) là biến Matlab eps, số lớn nhất được trình bày (*realmax* trong Matlab) và số dương nhỏ nhất được trình bày (*realmin* trong Matlab). Chạy chương trình Matlab đơn giản được cho ở file **NVD3_mparameters.m** trong Phụ lục 3A, nhận được các giá trị của một trong ba tham số quan trọng trên.

Chạy **NVD3_mparameters** theo tiêu chuẩn dấu phẩy động IEEE cho kết quả sau:

```
>> NVD3_mparameters
```

Giá trị của **isieee** là: 1

Giá trị của **esp**: 2.22044604925031e-16

Giá trị của ***realmax*** là: 1.7976931348623e+308

Giá trị của ***realmin*** là: 2.22500738585072e-308

Lưu ý rằng, khi chạy chương trình mô phỏng các dấu chấm trong các kết quả cũng như biểu diễn con số là dấu phẩy của tiếng Việt và ngược lại.

Kết quả đầu tiên (***isieee*** = 1) cho thấy máy tính thực hiện theo chuẩn ANSI/IEEE cho số dấu phẩy động. Kết quả tiếp theo là *eps*, là sai khác khả phân giải nhỏ nhất giữa hai số. Lưu ý, ***eps*** là 2^{-52} (bit lớn nhất liên quan đến phần định trị giải thích cho bit dấu), nó minh họa cho quan hệ giữa ***eps*** và chiều dài từ. Ta thấy rằng đạt được độ chính xác lớn hơn 15 con số. Nó là giá trị của ***eps*** quan hệ chặt chẽ với độ rộng của mức lượng tử Δ mà đã xét với số học dấu phẩy tĩnh. Lưu ý $\pm$ *realmax* xác định dải động.

Ví dụ 3.2: Giả sử ta sử dụng số học dấu phẩy động theo chuẩn ANSI/IEEE, để tính giá trị của:

$$A = 1 - 0{,}4 - 0{,}3 - 0{,}2 - 0{,}1 \tag{3.41}$$

Hiển nhiên kết quả là bằng không. Tuy nhiên, thực hiện phép tính này trong Matlab như sau:

```
>>    a = 1-0,4-0,3-0,2-0,1
      a = -2.7756e-017
```

Ta thấy rằng lỗi do dấu phẩy động nhỏ và có thể bỏ qua trong hầu hết các ứng dụng. Tuy nhiên, do lỗi khác 0, nên cần phải nhớ rằng kết quả tính toán không hoàn toàn chính xác.

3.4. Tái tạo và nội suy

Tại đây, ta xét việc khôi phục tín hiệu liên tục từ một chuỗi các mẫu. Trong mô phỏng số, do chỉ xử lý các giá trị mẫu, nên không thể khôi phục hoàn toàn tín hiệu liên tục từ một tập các mẫu. Tuy nhiên, khi xét quá trình khôi phục dẫn đến nội suy, là một hoạt động quan trọng trong mô phỏng.

Ở dạng tổng quát, khôi phục là cho các mẫu qua bộ lọc thông thấp có đáp ứng xung kim *h(t)*. Theo đó, dạng sóng khôi phục $x_r(t) = x_s(t) \otimes h(t)$, trong đó ⊗ ký hiệu cho tích chập. Từ (3.1) và (3.9) ta có:

$$x_s(t) = x(t)\sum_{k=-\infty}^{\infty}\delta\left(t - kT_s\right) = \sum_{k=-\infty}^{\infty}x\left(kT_s\right)\delta\left(t - kT_s\right) \tag{3.42}$$

Do vậy, dạng sóng khôi phục là:

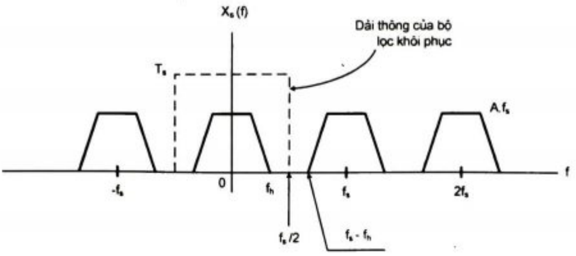
$$x_r(t) = \left[\sum_{k=-\infty}^{\infty}x\left(kT_s\right)\delta\left(t - kT_s\right)\right]\otimes h(t) \tag{3.43}$$

$$x_r(t) = \sum_{k=-\infty}^{\infty}x\left(kT_s\right)h\left(t - kT_s\right) \tag{3.44}$$

Vấn đề là lựa chọn h(t) sao cho các kết quả phù hợp ở một mức độ phức tạp tính toán hợp lý.

3.4.1. Tái tạo lý tưởng

Giả sử lấy mẫu một tín hiệu hạn chế băng tại tốc độ lớn hơn $2f_h$, thì có thể khôi phục tín hiệu bằng cách cho các mẫu qua bộ lọc thông thấp lý tưởng có độ rộng băng là $f_s/2$ (xem hình 3.10). Nếu $f_s > 2f_h$, thì phổ được trung tâm tại $f = \pm f_s$ không chồng lấn lên phổ được trung tâm tại $f = 0$. Tín hiệu ra của bộ lọc khôi phục là $f_s X(f)$ hay trong miền thời gian là $f_s x(t)$, lấy tỉ lệ biên độ bởi $1/f_s = T_s$ nhận được $x(t)$.



Hình 3.10: Bộ lọc khôi phục tín hiệu

Từ hình 3.10, đáp ứng xung kim của bộ lọc khôi phục là:

$$h(t) = T_s \int_{-f_s/2}^{f_s/2} e^{j2\pi ft} df \tag{3.45}$$

Trong đó tính đến hệ số tỉ lệ T_s. Vì vậy:

$$h(t) = T_s \frac{1}{j2\pi t}\left[e^{j\pi f_s t} - e^{j\pi f_s t}\right] \tag{3.46}$$

$$h(t) = T_s \frac{1}{\pi t}\sin\left(\pi f_s t\right) = \sin c\left(f_s t\right) \tag{3.47}$$

Thế vào (3.44) ta được:

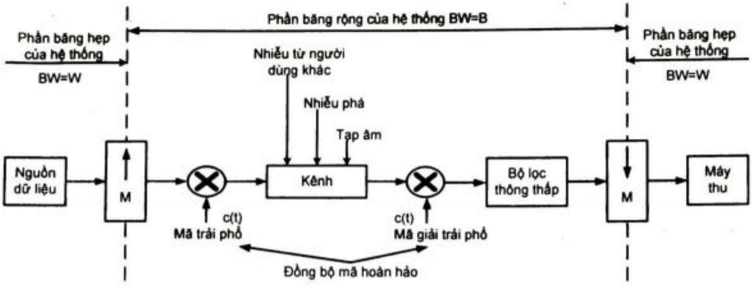
$$x_r(t) = \sum_{k=-\infty}^{\infty} x\left(kT_s\right)\sin c\left[f_s\left(t - kT_s\right)\right] \tag{3.48}$$

$$x_r(t) = \sum_{k=-\infty}^{\infty} x\left(kT_s\right)\sin c\left(\frac{t}{T_s} - k\right) \tag{3.49}$$

Lưu ý rằng, do giả định tín hiệu $x(t)$ bị hạn chế băng và tần số lấy mẫu đủ lớn để tránh được lỗi chồng phổ, nên $x_r(t) = x(t)$. Vì vậy, ít nhất về mặt lý thuyết đạt được khôi phục hoàn hảo. Tuy nhiên, (3.49) có thể không bao giờ được dùng trong thực tế, vì hàm *sinc(.)* là một hàm vô hạn về mặt phạm vi. Tuy vậy, phương trình (3.49) sẽ được dùng như là một cơ sở cho kỹ thuật nội suy thực tế ở phần sau.

3.4.2. Tăng mẫu và giảm mẫu

Tăng và giảm mẫu (upsampling và downsampling) được sử dụng trong quá trình mô phỏng nhiều hệ thống. Sự cần thiết được minh họa bằng ví dụ sau. Xét hệ thống trải phổ chuỗi trực tiếp hình 3.11. Khối nguồn dữ liệu tạo ra tín hiệu số liệu có phổ hẹp với băng tần là W. Tín hiệu số liệu được nhân với mã trải phổ băng rộng $c(t)$, mã này được biểu diễn bởi một chuỗi nhị phân có tốc độ lớn hơn nhiều tốc độ số liệu. Tỉ số giữa tốc độ mã trải phổ và tốc độ số liệu được gọi là độ lợi xử lý của hệ thống. Việc nhân với mã trải phổ $c(t)$ tạo ra một tín hiệu băng rộng có băng tần là B. Sự không hoàn hảo của kênh bao gồm nhiễu từ các người dùng khác, nhiễu phá (từ các hệ thống quân sự), tạp âm và một vài suy giảm khác như được minh họa trong hình 3.11. Tín hiệu đầu ra của kênh được nhân với mã giải trải phổ. Mã trải phổ nhận các giá trị ± 1, nếu mã trải phổ và mã giải trải phổ là giống nhau (đồng bộ chính xác), thì khi nhân mã trải phổ và mã giải trải phổ cho ta $c^2(t) = 1$, do vậy các mã trải phổ và giải trải phổ sẽ không ảnh hưởng lên kênh. Lưu ý rằng, tín hiệu số liệu được nhân với $c(t)$ hai lần và sự không hoàn hảo của kênh chỉ nhân với $c(t)$ một lần. Vì vậy, tại đầu vào của bộ lọc thông thấp ngay sau khi nhân tín hiệu với mã giải trải phổ, tín hiệu số liệu lại là tín hiệu băng hẹp và các thành phần khác là băng rộng. Bộ lọc thông thấp sẽ tách lấy thành phần băng hẹp và đưa nó đến máy thu.



Hình 3.11: Minh họa tính hữu hiệu của việc dùng tăng và giảm mẫu

Thuộc tính quan trọng của hệ thống được cho ở hình 3.11 là xuất hiện cả hai loại tín hiệu băng hẹp và băng rộng. Khi $B >> W$, lấy mẫu tín hiệu băng hẹp tại tần số lấy mẫu cho tín hiệu băng rộng sẽ không hiệu quả và kéo dài thời gian mô phỏng. Vì vậy, về lý tưởng mỗi tín hiệu được lấy mẫu tại tần số lấy mẫu phù hợp.

Do các tín hiệu có hai độ rộng băng tần khác nhau cùng tồn tại trong hệ thống, nên sử dụng hai tần số lấy mẫu. Do vậy, tần số lấy mẫu phải *tăng* tại biên giới giữa phần tín hiệu băng hẹp và băng rộng (đường đứt nét bên trái hình 3.11), và *giảm* tại biên giới giữa phần băng rộng và băng hẹp (đường đứt nét bên phải hình 3.11). Tăng tốc độ lấy mẫu được thực hiện bằng cách tăng mẫu theo sau là phép nội suy, ở đó các giá trị mẫu mới được nội suy từ các giá trị mẫu đã có. Việc giảm tốc độ lấy mẫu được thực hiện bằng cách loại bỏ mẫu có chọn lọc, ở đó các mẫu không cần thiết bị loại bỏ. Tăng mẫu được thể hiện bởi một khối có mũi tên hướng lên, và giảm mẫu được thể hiện bởi một khối có mũi tên hướng xuống. Tham số M thể hiện hệ số mà chu kỳ lấy mẫu giảm (tăng mẫu) hay tăng (giảm mẫu).

Phần sau, ta sẽ sử dụng T_s để thể hiện chu kỳ lấy mẫu trước khi quá trình tăng mẫu và giảm mẫu. Sau khi tăng mẫu hay giảm mẫu chu kỳ lấy mẫu sẽ được ký hiệu là T_u hay T_d. Tín hiệu trước khi tăng mẫu và giảm mẫu được ký hiệu là $x(t)$ (không có chỉ số dưới cho x), và tín hiệu sau khi tăng mẫu và giảm mẫu sẽ được ký hiệu là $x_u(t)$ và $x_d(t)$.

Tăng mẫu và nội suy

Tăng mẫu được minh họa trong hình 3.11 là quá trình thông qua nó tần số lấy mẫu được tăng lên. Vì tăng mẫu làm giảm chu kỳ lấy mẫu M lần, nên quan hệ giữa chu kỳ lấy mẫu mới T_u và chu kỳ lấy mẫu cũ T_s là $T_u = T_s/M$. Do vậy, với tín hiệu liên tục $x(t)$, thì quá trình tăng mẫu tạo ra các giá trị mẫu *mới* $x(kT_u) = x(kT_s/M)$ từ các giá trị lấy mẫu *cũ* $x(kT_s)$. Ví dụ: giả sử ta tạo một tập các mẫu *mới* bằng cách nội suy tín hiệu khôi phục $x_r(t)$ được cho bởi (3.49) tại các thời điểm $t = nT_s / M$. Thực hiện quá trình này ta có:

$$x_r\left(nT_u\right) = x\left(nT_s / M\right) = \sum_{k=-\infty}^{\infty} x\left(kT_s\right)\operatorname{sinc}\left(\frac{n}{M} - k\right) \tag{3.50}$$

Đây là một bộ nội suy không thể thực hiện được vì hàm *sinc(.)* mở rộng đến vô cùng. Cắt bớt hàm *sinc(.)* ta có:

$$x_i(nT_u) = x(nT_s / M) \cong \sum_{k=-L}^{L} x(kT_s)\sin C\left(\frac{n}{M} - k\right) \quad (3.51)$$

khả thi hơn, mặc dù không hoàn hảo. Hiển nhiên nếu L lớn sẽ giảm lỗi của nội suy. Tuy nhiên, vì mỗi tín hiệu được nội suy cần có $2L+1$ mẫu nên việc tính toán sẽ không chấp nhận được khi L lớn. Do vậy, cần phải dung hòa giữa mức độ chính xác và khối lượng tính toán khi nghiên cứu mô phỏng. Lưu ý rằng, do hàm nhân quả phải được dùng để nội suy nên gây ra một lượng trễ là LT_s. Trễ này không gây ra lỗi trong mô phỏng nhưng cũng cần phải lưu ý đến sự tồn tại của nó.

Một bộ nội suy thực tế hơn, yêu cầu tính toán ít hơn so với bộ nội suy hàm *sinc(.)* là *bộ nội suy tuyến tính*. Mặc dù, bộ nội suy tuyến tính đơn giản hơn bộ nội suy hàm *sinc(.)* nhưng vẫn có thể sử dụng khi tín hiệu được lấy mẫu quá lớn. Đáp ứng xung kim của bộ nội suy tuyến tính được cho bởi:

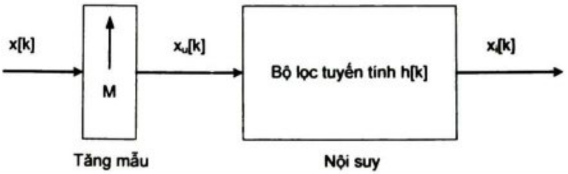
$$h[k] = \begin{cases} \dfrac{M - |k|}{M}, & k = 0, \pm 1, \pm 2, ..., \pm(M-1) \\ 0, & \textit{nếu khác} \end{cases} \quad (3.52)$$

Lưu ý rằng, có $2M-1$ giá trị khác không của $h[k]$. Chương trình Matlab để triển khai $h[k]$ được cho ở file **NVD3lininterp.m** dưới đây (có trong Phụ lục 3A).

Thực hiện tăng mẫu lên một tập các mẫu rời rạc là quá trình hai bước được minh họa ở hình 3.12. Trước hết ta tạo $x_u[k]$ từ $x[k]$ theo:

$$x_u[k] = \begin{cases} x[k/M], & k = 0, \pm M, \pm 2M, ..., \\ 0, & \textit{nếu khác} \end{cases} \quad (3.53)$$

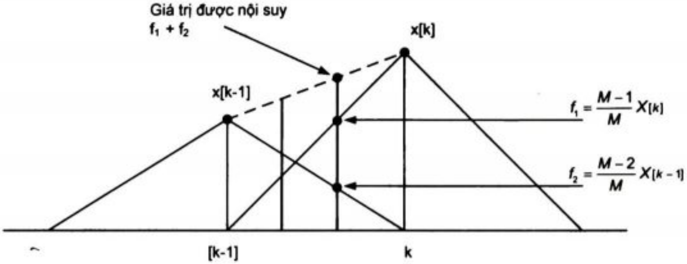
nó được thực hiện bằng chương trình Matlab ở file **NVD3_upsample.m** trong Phụ lục 3A.



Hình 3.12. Tăng mẫu và nội suy

Kết quả là, đặt $M-1$ mẫu giá trị 0 giữa mỗi mẫu trong chuỗi gốc $x[k]$. Sau đó thực hiện nội suy bằng cách lấy tích chập $x_u[k]$ với đáp ứng xung kim của bộ nội suy tuyến tính $h[k]$. Quá trình nội suy tuyến tính với $M=3$ được minh họa ở hình 3.13. Lưu ý rằng, chỉ cần sử

dụng 2 mẫu trong quá trình tăng mẫu. Trễ cần thiết là T_s. Như được minh họa ở hình 3.13, tìm được giá trị nội suy bằng cách lấy tổng các thành phần từ $x[k]$ và $x[k-1]$, đó là $\frac{M-1}{M}x[k]$ và $\frac{M-2}{2}x[k-1]$. Vì vậy, với $M = 3$, giá trị nội suy là:

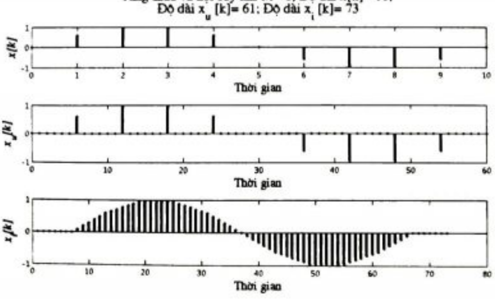


Hình 3.13: Minh họa quá trình nội suy

$$\frac{2}{3}x[k]+\frac{1}{3}x[k-1]$$

Vì chỉ có 2 mẫu được sử dụng trong quá trình nội suy, nên nội suy tuyến tính rất nhanh.

Ví dụ 3.3: Để minh họa quá trình tăng mẫu và nội suy, ta xét nội suy các mẫu của sóng hình sin. Các mẫu cơ bản được cho ở phần trên của hình 3.14 là $x[k]$. Tăng mẫu với $M = 6$ nhận được các giá trị mẫu $x_u[k]$. Nội suy tuyến tính với $M = 6$ nhận được chuỗi các mẫu $x_i[k]$. Lưu ý trễ T_x. Chương trình Matlab cho bài toán này được cho ở file **NVD3_upsampex.m** trong Phụ lục 3A, và kết quả mô phỏng điển hình được cho ở hình 3.14.



Hình 3.14: Nội suy và tăng mẫu được sử dụng trong ví dụ 3.3

Thấy rõ rằng, tăng mẫu và giảm mẫu bao gồm một lượng lớn mào đầu. Nếu hệ số tăng mẫu M là vừa phải, 2 hay 3, thì tốt nhất là triển khai mô phỏng bằng cách dùng một tần số lấy mẫu và do vậy xảy ra việc lấy mẫu quá nhiều không cần thiết đối với các tín hiệu băng hẹp. Tuy nhiên, nếu sai khác nhau giữa B và W lớn, thì hiệu quả hơn hết là sử dụng nhiều tần số lấy mẫu trong mô phỏng và lấy mẫu mỗi tín hiệu ở một tần số lấy mẫu phù hợp.

Giảm mẫu

Giảm mẫu là quá trình thông qua nó tần số lấy mẫu được giảm bằng cách thay một khối M mẫu bởi một mẫu. Theo đó, thực hiện giảm mẫu đơn giản hơn so với tăng mẫu rất nhiều. Thể hiện chức năng đối với các mẫu tại đầu ra của bộ giảm mẫu bằng cách thừa nhận rằng quá trình giảm mẫu làm tăng chu kỳ lấy mẫu lên M lần. Vì vậy, các mẫu ở đầu ra của bộ giảm mẫu là $x_d(kT_d) = x(kMT_s)$. Các giá trị lấy mẫu là

$$x_d[k] = x[kM] \tag{3.54}$$

Tuy nhiên, cần xét cẩn thận để đảm bảo tín hiệu được lấy mẫu thấp không gây ra chồng phổ.

3.5. Tần số lấy mẫu trong mô phỏng

Quyết định cơ bản phải được thực hiện trong quá trình triển khai mô phỏng là lựa chọn tần số lấy mẫu. Đối với hệ thống tuyến tính không hồi tiếp, ta phải chọn tần số lấy mẫu sao cho lỗi chồng phổ ở mức cho phép, nó phụ thuộc vào mật độ phổ công suất của dạng xung tín hiệu cơ bản. Vì vậy, ta xét mô hình để biểu diễn tín hiệu băng tần cơ sở, được dùng để truyền dẫn số liệu và để triển khai các kỹ thuật tính mật độ phổ công suất. Vì các dạng xung có vai trò quan trọng để chọn tần số lấy mẫu phù hợp, nên ta xét vấn đề này chi tiết hơn.

Từ việc nghiên cứu lấy mẫu, cho thấy để loại bỏ hoàn toàn lỗi chồng phổ cần có tần số lấy mẫu vô cùng lớn dẫn đến không thực tế. Ngoài ra, khi tần số lấy mẫu tăng, phải xử lý nhiều mẫu hơn cho mỗi ký hiệu dữ liệu, làm tăng thời gian mô phỏng. Vì không thể loại bỏ được hoàn toàn lỗi chồng phổ trong thực tế, nên ta phải chọn tần số lấy mẫu sao cho dung hòa giữa lỗi chồng phổ và thời gian mô phỏng. Tất nhiên, tần số lấy mẫu phải được chọn sao cho bỏ qua được lỗi chồng phổ khi so sánh với sự suy giảm hệ thống đang được xét bằng mô phỏng.

3.5.1. Khai triển chung

Mô hình chung cho tín hiệu phát trong hệ thống truyền tham số là:

$$x(t) = A\sum_{k=0}^{\infty} a_k p(t - kT - \Delta) \tag{3.55}$$

Trong đó: $\ldots, a_{-2}, a_{-1}, a_0, a_1, a_2, \ldots, a_k, \ldots$ là chuỗi các biến ngẫu nhiên biểu diễn số liệu. Giá trị a_k thường nhận giá trị +1 hay -1 trong hệ thống số nhị phân, $p(t)$ là hàm định dạng xung. T là chu kỳ ký hiệu, Δ là biến ngẫu nhiên phân bố đều trong khoảng chu kỳ lấy mẫu. Tham số A là một hằng số tỉ lệ để xác lập công suất của tín hiệu phát. Bằng cách kết hợp tham số này ta có thể lấy tỉ số hàm định dạng xung sao cho giá trị đỉnh là đơn vị. Giả sử $E\{a_k\} = 0$ và $E\{a_k a_{k+m}\} = R_m$ là giá trị trung bình và tự tương quan của chuỗi xung số liệu.

Dễ thấy rằng hàm tự tương quan của tín hiệu phát được cho bởi:

$$R_{xx}(\tau) = A^2 \sum_{m=-\infty}^{\infty} R_m . r(\tau - mT) \tag{3.56}$$

Trong đó:

$$r(\tau) = \frac{1}{T} \int_{-\infty}^{\infty} p(t)p(t+\tau)dt \tag{3.57}$$

Tần số lấy mẫu được xác định từ mật độ phổ công suất PSD của tín hiệu phát. Áp dụng định lý Weiner-Khintchine vào (3.56) ta có:

$$S_X(f) = A^2 \int_{-\infty}^{\infty} \left(\sum_{m=-\infty}^{\infty} R_m r(\tau - mT) \right) e^{-j2\pi f\tau} d\tau \tag{3.58}$$

$$S_X(f) = A^2 \sum_{m=-\infty}^{\infty} R_m \int_{-\infty}^{\infty} r(\tau - mT) e^{-j2\pi f\tau} d\tau \tag{3.59}$$

Ta biến đổi về dạng dễ tính toán hơn.

Trước tiên đổi biến số $\alpha = \tau - mT$ vào (3.59). Ta có:

$$S_X(f) = A^2 \sum_{m=-\infty}^{\infty} R_m \int_{-\infty}^{\infty} r(\alpha) e^{-j2\pi f(\alpha + mT)} d\alpha \tag{3.60}$$

Ký hiệu:

$$S_r(f) = \int_{-\infty}^{\infty} r(\alpha) e^{-j2\pi f\alpha} d\alpha \tag{3.61}$$

Ta có:

$$S_X(f) = A^2 \sum_{m=-\infty}^{\infty} R_m . S_r(f) . e^{-j2\pi fmT} \tag{3.62}$$

Ta phải xác định $S_r(f)$

Biến đổi Fourier lên (3.57) ta có:

$$S_r(f) = \int_{-\infty}^{\infty} \left(\frac{1}{T} \int_{-\infty}^{\infty} p(t)p(t+\alpha)dt \right) . e^{-j2\pi f\alpha} d\alpha \tag{3.63}$$

Đổi biến $\beta = t + \alpha$ thì (3.63) có thể biểu diễn ở dạng như sau:

$$S_r(f) = \frac{1}{T} \left(\int_{-\infty}^{\infty} p(t).e^{j2\pi ft} dt \right) . \left(\int_{-\infty}^{\infty} p(\beta).e^{-j2\pi f\beta} d\beta \right) \tag{3.64}$$

Thành phần thứ hai trong (3.64) là biến đổi Fourier của hàm dạng xung $p(t)$ và thành phần đầu tiên là liên hợp phức của thành phần thứ hai. Và ta có:

$$S_r(f) = \frac{|P(f)|^2}{T} = \frac{G(f)}{T} \quad (3.65)$$

Trong đó $G(f)$ là mật độ phổ *năng lượng* của hàm định dạng xung $p(t)$. Thế (3.65) vào (3.62) cho ta kết quả tổng quát như sau:

$$S_X(f) = A^2 \frac{G(f)}{T} \sum_{m=-\infty}^{\infty} R_m \, e^{-j2\pi f mT} \quad (3.66)$$

Trong nhiều ứng dụng các ký hiệu số liệu có thể coi là độc lập. Giả thiết trên dẫn đến đơn giản hóa rất nhiều.

3.5.2. Ký hiệu dữ liệu độc lập

Nếu các ký hiệu dữ liệu $\{a_k\}$ là độc lập, thì hàm tự tương quan trở thành:

$$\begin{aligned} R_m &= E\{a_k a_{k+m}\} = E\{a_k\} E\{a_{k+m}\} \\ &= a_k^2 \delta[m] = \begin{cases} a_k^2, & m = 0 \\ 0, & \text{nếu khác} \end{cases} \end{aligned} \quad (3.67)$$

Khi đó, mật độ phổ công suất PSD của $x(t)$ được định nghĩa bởi (3.66) trở nên rất đơn giản:

$$S_X(f) = A^2 \frac{a_k^2 G(f)}{T} \quad (3.68)$$

Nếu các ký hiệu số liệu $a_k = \pm 1 \; \forall k \Rightarrow a_k^2 = 1$ và:

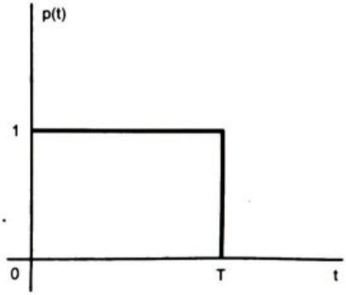
$$S_X(f) = A^2 \frac{G(f)}{T} \quad (3.69)$$

không phụ thuộc vào số liệu. Trường hợp các ký hiệu dữ liệu không độc lập, phải xác định hàm tự tương quan R_m và (3.66) phải được tính toán từng phần một.

Ví dụ 3.4: Xét dạng xung chữ nhật được cho ở hình 3.15. Theo hình 3.15 ta có:

$$P(f) = \int_0^T e^{-j2\pi f t} dt = \frac{1}{j2\pi f}\left[1 - e^{-j2\pi f T}\right] \quad (3.70)$$

$$\begin{aligned} P(f) &= \frac{1}{j2\pi f}\left[e^{j\pi f T} - e^{-j\pi f T}\right].e^{-j\pi f T} \\ &= \frac{\sin(\pi f T)}{\pi f} e^{-j\pi f T} \end{aligned} \quad (3.71)$$



Hình 3.15: Dạng xung chữ nhật

Hoặc ở dạng hàm *sinc(.)*:

$$P(f) = T\sin c\left(fT\right).e^{-j\pi fT} \tag{3.72}$$

Do vậy: $G(f) = \left|P(f)\right|^2 = T^2 \sin c^2(fT)$

Thế vào (3.69) ta có mật độ phổ công suất của $x(t)$ là:

$$S_X(f) = A^2 T \sin c^2(fT) \tag{3.73}$$

Công suất phát từ (3.69):

$$P = \int_{-\infty}^{\infty} S_X(f)df = A^2 \frac{1}{T}\int_{-\infty}^{\infty} G(f)df \tag{3.74}$$

Theo định lý Parseval's và hình 3.15 ta có:

$$\int_{-\infty}^{\infty} G(f)df = \int_{-\infty}^{\infty}\left|P(f)\right|^2 df = \int_{-\infty}^{\infty} p^2(t)dt = T \tag{3.75}$$

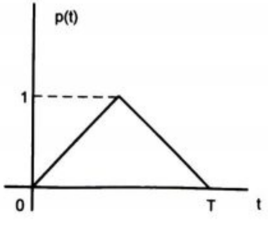
Thế vào (3.74) ta có:

$$P = A^2 \tag{3.76}$$

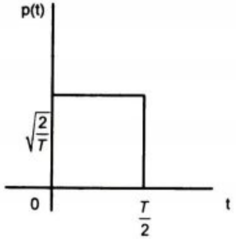
Kết quả đơn giản này xuất phát từ thực tế là $p(t)$ là một xung đơn vị. Do vậy, $\sum a_k p(t-kT-\Delta)$ có công suất đơn vị. Phép nhân với A như trong (3.55) chỉ là lấy tỉ lệ công suất bởi A^2. Với các dạng xung khác quan hệ giữa công suất và A phải được tính toán theo kỹ thuật đã minh họa ở trên. Lưu ý rằng chuỗi số liệu $\{a_k\}$ là quá trình công suất (phương sai) đơn vị.

Ví dụ 3.5: Dạng xung được mô tả trong hình 3.16(a). Dạng xung cơ bản $p(t)$ được biểu diễn dưới dạng $p_1(t)\otimes p_1(t)$ trong đó $\otimes$ là tích chập, và $p_1(t)$ được mô tả trong hình 3.16(b). Lấy biến đổi Fourier của $p_1(t)$ ta có:

$$P_1(f) = \sqrt{\frac{T}{2}}\sin c\left(\frac{T}{2}f\right).e^{-j\pi fT/2} \tag{3.77}$$



a) Dạng xung tam giác



b) Dạng xung cơ sở để lấy tích chập

Hình 3.16: Dạng xung tam giác

Vì phép tích chập trong miền thời gian tương đương phép nhân trong miền tần số, nên:

$$|P(f)| = |P_1(f)|^2 = \frac{T}{2}\sin c^2\left(\frac{T}{2}f\right) \tag{3.78}$$

Do vậy:

$$G(f) = |P(f)|^2 = \frac{T^2}{4}\sin c^4\left(\frac{T}{2}f\right) \tag{3.79}$$

Thay vào (3.69) ta có:

$$S_X(f) = \frac{A^2T}{4}\sin c^4\left(\frac{T}{2}f\right) \tag{3.80}$$

Kết quả này sẽ được sử dụng ở chương 4.

3.5.3. Tần số lấy mẫu trong mô phỏng

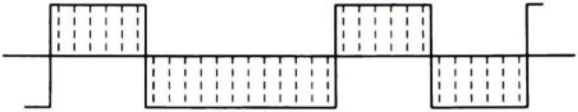
Trở lại vấn đề tần số lấy mẫu mô phỏng dạng xung bằng cách xét SNR của quá trình lấy mẫu trong đó công suất tạp âm do chồng phổ. Với mục đích chọn tần số lấy mẫu sao cho có thể bỏ qua lỗi chồng phổ so với suy giảm chất lượng hệ thống được mô phỏng. Có thể thấy rằng tần số lấy mẫu phụ thuộc vào dạng sóng mô hình mô phỏng.

Xét dạng sóng được cho bởi (3.55) có dạng xung chữ nhật được minh họa ở hình 3.15, được lấy mẫu như trong hình 3.17.

Xác định mật độ phổ công suất của tín hiệu

Trong hình 3.17, tần số lấy mẫu gấp 6 lần tốc độ ký hiệu. Mật độ phổ công suất của chuỗi ký hiệu được cho bởi (3.73). Kết hợp kết quả này với (3.18) ta có:

$$S_{X_s}(f) = f_s^2 \sum_{n=-\infty}^{\infty} A^2T\sin c^2\left[(f - nf_s)T\right] \tag{3.81}$$



Hình 3.17: Chuỗi xung chữ nhật được lấy mẫu tại 6 mẫu/ký hiệu

Lấy mẫu chuỗi số liệu tại tốc độ m mẫu trên một ký hiệu $(f_s = m/T)$ ta có:

$$S_{X_s}(f) = f_s^2 \sum_{n=-\infty}^{\infty} A^2T\sin c^2\left[\left(f - \frac{nm}{T}\right)T\right] \tag{3.82}$$

$$S_{X_s}(f) = f_s^2 \sum_{n=-\infty}^{\infty} A^2T\sin c^2\left[fT - nm\right] \tag{3.83}$$

Tính tỉ số tín hiệu trên tạp âm do chồng phổ

Tỉ số tín hiệu trên tạp âm do chồng phổ $(SNR)_a = S/N_a$ trong đó:

Công suất tín hiệu là:

$$S = \int_{-f_s/2}^{f_s/2} f_s^2 A^2 T \operatorname{sinc}^2(fT)df = 2f_s^2 A^2 T \int_0^{f_s/2} \operatorname{sinc}^2(fT)df \qquad (3.84)$$

Công suất tạp âm do chồng phổ là:

$$\begin{aligned} N_a &= \int_{-f_s/2}^{f_s/2} f_s^2 \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} A^2 T \operatorname{sinc}^2(fT - nm)df \\ &= 2f_s^2 A^2 T \sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \int_0^{f_s/2} \operatorname{sinc}^2(fT - nm)df \end{aligned} \qquad (3.85)$$

Lưu ý rằng, PSD là hàm chẵn của tần số. Công suất tín hiệu được xác định bởi tích phân thành phần $n = 0$ trong (3.83) trên dải tần mô phỏng $|f| < f_s/2$. Công suất tập âm do chồng phổ là công suất từ tất cả các tần số ($n \neq 0$) trong dải tần mô phỏng. Do vậy, ta tìm được tạp âm do chồng phổ bằng cách lấy tích phân trên tất cả các thành phần trong (3.83) trừ thành phần $n = 0$. Thấy rõ trong hình 3.18, được vẽ với $m = 6$. Hình 3.18 minh họa phần tần số dương của thành phần $n = 0$ trong (3.83) trên dải $0 < f < f_s$. Cho thấy, dịch phổ đối với $n = \pm 1$ và $n = 2$.

Bước tiếp theo trong quá trình xác định $(SNR)_a$, phải chỉ ra rằng cả S và N_a đều có thể xác định được bằng cách chỉ sử dụng thành phần $n = 0$ trong (3.83). Các *búp* của hàm *sinc(.)* đều có độ rộng 1/T, được minh họa và đánh số trong hình 3.18. Lưu ý rằng với $m = 6$ và $n = 0$, các búp 1,2 và 3 rơi vào trong khoảng $0 < f < f_s/2$ vì vậy thể hiện công suất tín hiệu. Các búp 4, 5 và 6 rơi vào trong khoảng $0 < f < f_s/2$ với thành phần $n = 1$ và do vậy nó thể hiện nhiễu chồng phổ. Tương tự như vậy, các búp 7, 8 và 9 từ thành phần $n = -1$ trong (3.83) và búp 10, 11 và 12 từ thành phần $n = 2$ trong (3.83). Tiếp tục như vậy ta có:

$$\sum_{\substack{n=-\infty \\ n\neq 0}}^{\infty} \int_0^{f_s/2} \operatorname{sinc}^2(fT - nm)df = \int_{f_s/2}^{\infty} \operatorname{sinc}^2(fT - nm)df \qquad (3.86)$$

Vì vậy:

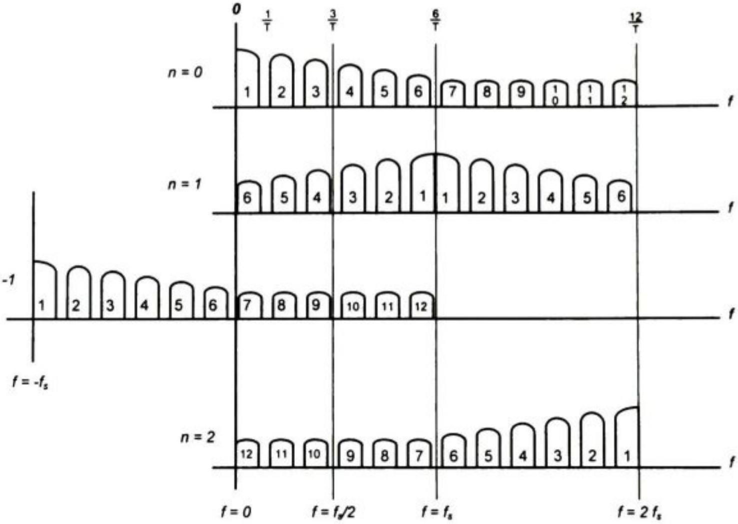
$$(SNR)_a = \frac{S}{N_a} = \frac{\int_0^{f_s/2} \operatorname{sinc}^2(fT)df}{\int_{f_s/2}^{\infty} \operatorname{sinc}^2(fT)df} \qquad (3.87)$$

Bằng cách *so sánh* các ví dụ 3.4 và 3.5 thấy rõ dạng của hàm bị tích phân sẽ khác nhau phụ thuộc vào dạng xung.

Ta sẽ thường xuyên tìm xem có cần thiết sử dụng tích phân số để định lượng $(SNR)_a$. Để thực hiện điều này, quá trình lấy mẫu thứ hai được thực hiện trong đó biến tần số liên tục f được lấy mẫu ở các điểm $f = jf_1$. Để chính xác thì $f_1 << 1/T$ do vậy rất nhiều mẫu được lấy ở mỗi

búp của hàm *sinc(.)*. Theo đó, lấy mẫu tần số cho phép tích phân trong (3.87) được thay thế bởi lấy tổng. Để thỏa mãn $f_1 << 1/T$ đặt $f_1 = 1/(kT)$ trong đó k phải lớn để lỗi do tích phân số là nhỏ. Với $f = jf_1$ và $f_1 = 1/(kT)$ ta có:

$$fT = \frac{j}{k} \tag{3.88}$$



Hình 3.18: Phổ và sự dịch chuyển phổ khi n = ±1 & n = ±2

Bước tiếp theo là tính toán tần số $f_s/2$ theo các tham số rời rạc k và m. Từ hình 3.18 cho thấy, với m mẫu trên một kí hiệu, thì tần số $f_s/2$ là $m/(2T)$. Vì lấy k mẫu cho khoảng tần số có độ rộng $1/T$, nên tần số tương ứng với chỉ số $km/2$. Sử dụng (3.88) và $f_s/2$ tương ứng $km/2$ trong (3.87) ta có:

$$(SNR)_a \cong \frac{\sum_{j=0}^{km/2} \operatorname{sinc}^2(j/k)}{\sum_{j=km/2}^{\infty} \operatorname{sinc}^2(j/k)} \tag{3.89}$$

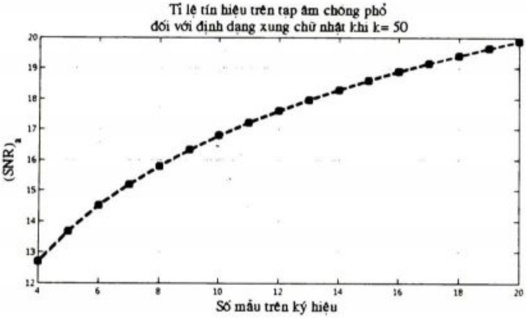
Có được xấp xỉ trên là vì sử dụng tích phân số để lấy xấp xỉ đến giá trị thực của tích phân.

Chương trình Matlab để ước tính (3.89) được cho bởi file **NVD3_sna.m** trong Phụ lục 3A.

Lưu ý rằng, lấy 50 mẫu tần số cho mỗi búp của hàm *sinc(.)* và có tổng số 50000 mẫu tần được lấy. Vì vậy, tổng ở mẫu số của (3.89) có đến 1000 bup của hàm *sinc(.)*, sau khoảng này PSD được cho là có thể bỏ qua. Có thể kiểm tra giả thiết này bằng cách làm thí nghiệm với tham số ***nsamp***.

Chạy chương trình mô phỏng **NVD3_sna.m,** kết quả được minh họa ở 3.19. Lưu ý rằng, $(SNR)_a$ là nhỏ hơn 17 dB với $m = 10$ mẫu trên một ký hiệu và $(SNR)_a$ sẽ tăng khi m tăng. Tuy nhiên, ảnh hưởng lên $(SNR)_a$ giảm khi m tăng. Cũng lưu ý rằng PSD của tín hiệu mẫu giảm $1/f^2$ với dạng xung hình chữ nhật. Ví dụ 3.5 cho thấy, PSD của tín hiệu mẫu giảm $1/f^4$ với dạng xung hình tam giác. Vì vậy, với mỗi giá trị cho trước của m, thì giá trị $(SNR)_a$ đối với dạng xung tam giác *lớn hơn* $(SNR)_a$ đối với dạng xung hình chữ nhật. Dạng xung hình chữ nhật là trường hợp tồi nhất.

Trong hệ thống truyền thông thực tế, dạng xung $p(t)$ được chọn để có được hiệu quả băng thông yêu cầu. Hiệu quả băng tần cao cho thấy phổ của $x(t)$ được cho bởi (3.55) là tập trung xung quanh thành phần $f = 0$. Do đó, các tín hiệu có hiệu quả băng thông cao cần có giá trị m nhỏ đối với mỗi giá trị $(SNR)_a$ đã cho.



Hình 3.19: Tỉ số tín hệu trên tạp âm chồng phổ đối với dạng xung chữ nhật

3.6. Điều chế xung mã PCM

3.6.1 Lượng tử hóa

Ở dạng tổng quát, các sơ đồ lượng tử hoá có thể được phân loại thành hai loại sau: *(i)* sơ đồ lượng tử hoá vô hướng, trong đó lại được phân thành lượng tử hóa đều và không đều; *(ii)* sơ đồ lượng tử hoá véc tơ. Trong lượng tử hoá vô hướng, mỗi đầu ra của nguồn tin được lượng tử hoá riêng rẽ, trong khi đó lượng tử hoá véc tơ, các khối đầu ra của nguồn tin được lượng tử hóa.

Trong lượng tử hóa đều, các vùng được chọn có cùng chiều dài, còn ở lượng tử hóa không đều, thì cho phép sử dụng các vùng có chiều dài có chiều dài khác nhau. Ở dạng tổng quát, các bộ lượng tử hoá không đều tốt hơn các bộ lượng tử hoá đều. Ở đây ta chỉ xét lượng tử hóa vô hướng mà không xét cho lượng tử hóa véc tơ.

Trong lượng tử hoá vô hướng, dải giá trị của biến ngẫu nhiên X được chia thành N vùng $\Re_i$ với $1 \le i \le N$ không chồng chất nhau, được gọi là các khoảng lượng tử hoá, và trong mỗi vùng ta chọn một điểm được gọi là một mức lượng tử. Khi đó, mọi giá trị của biến ngẫu nhiên mà rơi vào vùng $\Re_i$ được lượng tử thành mức lượng tử thứ i, được ký hiệu là $\hat{x}_i$. Nghĩa là:

$$x \in \Re_i \Leftrightarrow Q(x) = \hat{x}_i \tag{3.90}$$

Trong đó:

$$\hat{x}_i \in \Re_i \tag{3.91}$$

Hiển nhiên, lượng tử hoá kiểu này tạo ra một ***sai số bình phương trung bình*** **MSE** là $E\left[\left(x-\hat{x}_i\right)^2\right]$. Vì vậy, sai số lượng tử bình phương trung bình được cho bởi

$$D = \sum_{i=1}^{N} \underbrace{\int_{\Re_i} \underbrace{\underbrace{\left(x-\hat{x}_i\right)}_{\text{sai số}}{}^2}_{\text{lấy bình phương}} f_X(x)dx}_{\substack{\text{lấy trung bình của biến ngẫu nhiên } (x-\hat{x}_i)^2 \\ \text{có pdf là } f_X(x) \text{ trên vùng } \Re_i}} \tag{3.92}$$

Trong đó $f_X(x)$ ký hiệu cho hàm mật độ xác suất pdf của biến ngẫu nhiên của nguồn tin. Tỷ số tín hiệu trên tạp âm lượng tử SQNR (Signal-to-Quantization-Noise Ratio) được định nghĩa là:

$$SQNR = 10\log_{10}\frac{E\left[X^2\right]}{D},\ dB$$

Lượng tử hoá đều

Trong quá trình lượng tử hoá đồng đều, tất cả các vùng lượng tử hóa ngoại trừ vùng đầu tiên và vùng cuối cùng, tức là vùng $\Re_1$ và $\Re_N$ đều có cùng độ dài được ký hiệu là Δ; vì vậy:

$$\begin{aligned}
\Re_1 &= \left(-\infty, a\right] \\
\Re_2 &= \left(a, a+\Delta\right] \\
\Re_3 &= \left(a+\Delta, a+2\Delta\right] \\
&\ldots \\
\Re_N &= \left(+(N-2)\Delta, \infty\right]
\end{aligned}$$

Mức lượng tử hoá tối ưu trong mỗi khoảng lượng tử hoá được chứng minh là trọng tâm của khoảng đó; nghĩa là:

$$\hat{x}_i = E\left[X \middle| X \in \Re_i\right] = \frac{\int_{\Re_i} x.f_X(x)dx}{\int_{\Re_i} f_X(x)dx}, \qquad 1 \le i \le N \tag{3.93}$$

Do đó, việc thiết kế bộ lượng tử hoá đều tương đương với việc xác định a và Δ. Một khi a và Δ đã được xác định, các giá trị của $\hat{x}_i$ và lượng méo dễ dàng được xác định bằng các biểu thức (3.92) và (3.93). Trong một số trường hợp, đơn giản ta chọn các mức lượng tử là các điểm giữa của các vùng lượng tử, nghĩa là chọn tại khoảng cách $\Delta/2$ tính từ các biên của các vùng lượng tử.

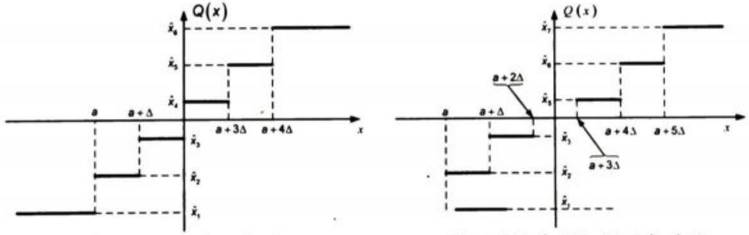
Đồ thị của hàm lượng tử hoá Q(x) đối với một hàm mật độ xác suất đối xứng của X và các giá trị chẵn và lẻ của N được thể hiện trên hình 3.20 tương ứng.

Khi hàm mật độ xác suất đối xứng, bài toán trở nên đơn giản hơn. Trong trường hợp như vậy.

$$\Re_i = \begin{cases} (a_{i-1}, a_i], & 1 \le i \le N-1 \\ (a_{i-1}, a_N), & 1 = N \end{cases} \tag{3.94}$$

Trong đó:

$$\begin{cases} a_0 = -\infty \\ a_i = (i - N/2)\Delta & 1 \le i \le N-1 \\ a_N = \infty \end{cases} \tag{3.95}$$



a) Lượng tử hóa đều với $N = 6$, lưu ý rằng $a + 2\Delta = 0$

b) Lượng tử hóa đều với $N = 7$, lưu ý rằng $\hat{x}_4 = 0$

Hình 3.20: Minh họa lượng tử hóa đồng đều

Ta thấy rằng, trong trường hợp này, ta chỉ có một tham số Δ phải được chọn để đạt được méo tối thiểu. Giả sử rằng, các mức lượng tử được đặt tại các trọng tâm của các vùng lượng tử hoá, ba chương trình Matlab **NVD3_centroid.m, NVD3_mse_dist.m, NVD3_uq_dist.m** trong Phụ lục 3A thực hiện tìm: *(i)* trọng tâm của một vùng; *(ii)* sai số lượng tử bình phương trung bình (MSE) khi cho trước một phân bố và các biên của vùng lượng tử hoá; *(iii)* lượng méo. Để sử dụng các hàm này, thì phân bố của nguồn (phụ thuộc đến 3 tham số) phải được cho trong một hàm.

Để được rõ hơn, các bài tập được viết trên Matlab để: *(i)* xác định trọng tâm của các vùng lượng tử hóa đối với phân bố Gausơ; *(ii)* xác định sai số trung bình bình phương đối với bộ lượng tử; *(iii)* khảo sát bộ lượng tử với các mức được đặt ở các điểm giữa của vùng. Tất cả được cho ở Phụ lục 3A và 3B.

Lượng tử hoá không đều

Trong lượng tử hoá không đều, yêu cầu về các vùng lượng tử hoá (ngoại trừ vùng đầu và cuối) có độ dài như nhau được xem nhẹ, và mỗi vùng lượng tử hóa có thể có một độ dài bất kỳ. Do trong trường hợp này việc tối ưu hoá được thực hiện với những điều kiện giảm nhẹ hơn, kết

quả là có sự vượt trội rõ ràng hơn so với lượng tử hoá đều. Các điều kiện tối ưu khi này, được gọi là các điều kiện Lloyd-Max, và được biểu diễn là:

$$\begin{cases} \hat{x}_i = \dfrac{\int_{a_{i-1}}^{a_i} x f_X(x)dx}{\int_{a_{i-1}}^{a_i} f_X(x)dx} \\ a_i = \dfrac{(\hat{x}_{i-1} - \hat{x}_i)}{2} \end{cases} \tag{3.96}$$

Từ những phương trình này, ta kết luận rằng các mức lượng tử tối ưu là các trọng tâm của các vùng lượng tử và các biên tối ưu giữa các vùng lượng tử là các điểm giữa của các mức lượng tử. Để có được lời giải cho các phương trình Lloyd-Max, ta bắt đầu với tập các mức lượng tử $\hat{x}_i$. Từ tập này, ta có thể tìm được một tập các biên vùng lượng tử a_i. Từ tập các a_i này, ta nhận được tập mới các mức lượng tử. Quá trình này được tiếp tục cho đến khi việc cải thiện về méo từ bước lặp này sang bước lặp khác là nhỏ. Thuật toán này bảo đảm hội tụ tới một giá trị cực tiểu cục bộ, nhưng ở dạng tổng quát thì khó đạt được giá trị tối thiểu toàn bộ.

Thủ tục thiết kế bộ lượng tử hoá tối ưu được trình bày trong file **NVD3_lloydmax.m** ở Phụ lục 3A. Bài tập minh họa trên Matlab cho vấn đề này được cho ở Phụ lục 3A và 3B.

3.6.2. Điều chế xung mã với lượng tử hóa đều và không đều

Trong điều chế xung mã, tín hiệu tương tự trước hết được lấy mẫu tại tốc độ lớn hơn tốc độ Nyquist, sau đó lượng tử hóa các mẫu này. Giả sử tín hiệu tương tự được phân bố trên khoảng $[-x_{max}, x_{max}]$ và số mức lượng tử là lớn. Các mức lượng tử có thể cách đều hoặc không cách nhau. Dưới đây ta khảo sát cho hai trường hợp này là PCM lượng tử hoá đều và PCM lượng tử hoá không đều.

PCM với lượng tử hoá đều

Trong trường hợp này, khoảng $[-x_{max}, x_{max}]$ có độ dài $2x_{max}$ được chia thành N khoảng con bằng nhau, mỗi khoảng con có độ dài $\Delta = 2x_{max}/N$. Nếu N đủ lớn, thì hàm mật độ xác suất của tín hiệu lối vào trên mỗi khoảng con có thể được coi là phân bố đều, dẫn đến một lượng méo $D = \Delta^2/12$. Nếu N là một luỹ thừa của 2 (tức là $N = 2^v$), thì cần có v bit để biểu diễn cho mỗi mức lượng tử. Nghĩa là, nếu độ rộng băng của tín hiệu tương tự là W, và nếu thực hiện lấy mẫu tại tốc độ Nyquist, thì độ rộng băng cần thiết để truyền dẫn tín hiệu PCM tối thiểu là vW (thực tế thường lấy $1,5vW$). Lượng méo được cho bởi:

$$D = \frac{\Delta}{12} = \frac{x_{max}^2}{3N^2} = \frac{x_{max}^2}{3 \times 4^v} \tag{3.97}$$

Nếu công xuất của tín hiệu tương tự được ký hiệu là $\overline{X^2}$, thì tỉ số tín hiệu trên tạp âm lượng tử SQNR được cho bởi:

$$SQNR = 3.v^2 \frac{\overline{X^2}}{x_{max}^2} = 3 \times 4^v \frac{\overline{X^2}}{x_{max}^2} = 3 \times 4^v \overline{\breve{X}^2} \tag{3.98}$$

Trong đó $\breve{X}$ là tín hiệu đầu vào chuẩn hoá được xác định là $\breve{X} = \frac{X}{X_{max}}$

SQNR tính theo đơn vị dB là:

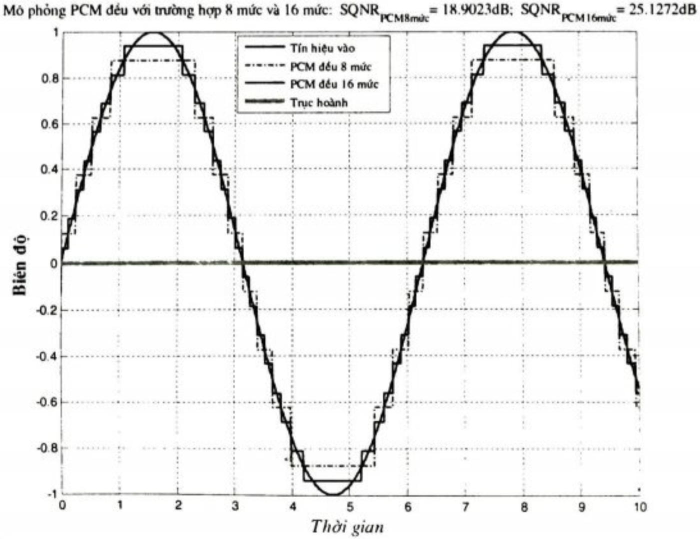
$$SQNR\big|_{dB} \approx 4,8 + 6v + \overline{\breve{X}^2}\big|_{dB} \tag{3.99}$$

Sau khi lượng tử hoá, mỗi mức lượng tử được mã hóa bởi v bit. Phương pháp mã hoá thường dùng là mã nhị phân tự nhiên (*NBC: Natural Binary Coding*), nghĩa là mức thấp nhất được ánh xạ thành một chuỗi toàn bit 0 và mức cao nhất được ánh xạ thành một chuỗi toàn bit 1. Toàn bộ các mức còn lại được ánh xạ theo thứ tự tăng dần của giá trị lượng tử.

Hàm **NVD3_u_pcm.m** trong Phụ lục 3A thực hiện bài toán này, theo đó nó nhận 2 tham số đầu vào là: chuỗi các giá trị mẫu a, số mức lượng tử hoá mong muốn; cho 3 kết quả đầu ra là: chuỗi lượng tử hóa, chuỗi bit đã mã hóa, và tỉ số tín hiệu trên tạp âm lượng tử SQNR theo đơn vị dB.

Ví dụ 3.6: [PCM lượng tử hoá đều]

Hãy tạo một tín hiệu hình sin có biên độ bằng 1 và tần số $\omega = 1$. Sử dụng sơ đồ PCM lượng tử hoá đều để: *(i)* lượng tử hoá 8 mức và 16 mức; *(ii)* vẽ đồ thị tín hiệu gốc và các tín hiệu lượng tử hoá trên cùng một hệ toạ độ; *(iii)* so sánh SQNR trong trường hợp 8 mức lượng tử và 16 mức lượng tử.



*Hình 3.21: **PCM lượng tử hóa đều** cho tín hiệu hìn sin dùng 8 mức và 16 mức lượng tử*

Ta chọn một cách tuỳ ý thời gian tồn tại của tín hiệu là 10s (thời gian mô phỏng). Sau đó, hàm **NVD3_u_pcm.m** trong Phụ lục 3A để tạo các tín hiệu lượng tử cho hai trường hợp 8 và 16 mức lượng tử. Các SQNR tính được là: 18,90 dB đối với PCM 8 mức lượng tử hoá đều và 25,13 dB đối với trường hợp PCM lượng tử hóa đều 16 mức. Chương trình mô phỏng thực hiện bài toán này được cho ở file **NVD3_Sim36.m** trong Phụ lục 3A. Kết quả chạy chương trình này được cho ở hình 3.21.

Ví dụ 3.7: PCM lượng tử hoá đều

Hãy tạo một chuỗi 500 biến ngẫu nhiên Gausơ trung bình 0 và phương sai bằng 1. Sử dụng hàm **NVD3_u_pcm.m** trong Phụ lục 3A để tìm SQNR khi số mức lượng tử hoá là 64. Hãy tìm 5 giá trị đầu tiên của chuỗi, tương ứng với các giá trị đã lượng tử , và các từ mã tương ứng.

Chương trình Matlab **NVD3_sim37.m** trong Phụ lục 3A thực hiện bài toán này.

Khi chạy chương trình **NVD3_sim37**, quan trắc được các kết quả sau:

$$\text{SQNR} = 31{,}66 \text{ dB}$$

Tín hiệu lối vào = [0,1775; -0,4540; 1,0683; - 2,2541; 0,5376]

Các giá trị sau lượng tử = [0,1569; -0,4708; 1,0985; -2,2494; 0,5754]

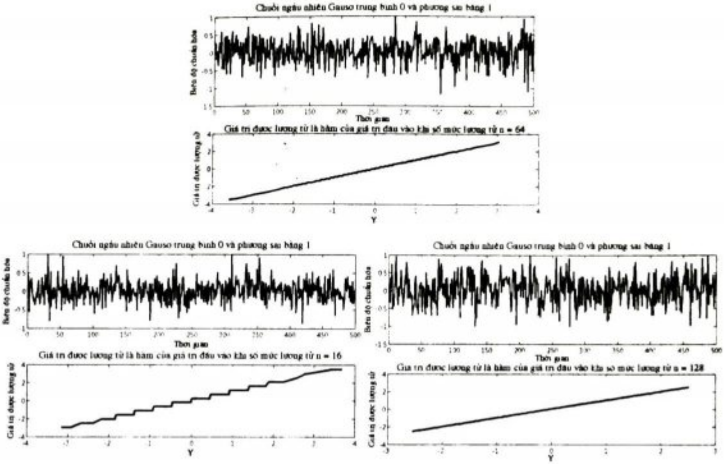
$$\text{Các từ mã } = \begin{cases} 1\ 0\ 0\ 0\ 0\ 1 \\ 0\ 1\ 1\ 0\ 1\ 1 \\ 1\ 0\ 1\ 0\ 1\ 0 \\ 0\ 0\ 1\ 0\ 1\ 0 \\ 1\ 0\ 0\ 1\ 0\ 1 \end{cases}$$

Lưu ý rằng, các lần chạy khác nhau của chương trình này cho ra các kết quả khác nhau đối với tín hiệu đầu vào, các giá trị sau lượng tử, và các từ mã. Tuy nhiên, các SQNR tính được đều rất sát nhau.

Ví dụ 3.8: Sai số lượng tử

Từ ví dụ 3.7, hãy vẽ đồ thị sai số lượng tử, được định nghĩa là sự khác nhau giữa giá trị đầu vào và giá trị được lượng tử hóa cho các trường hợp 16, 64, và 128 mức lượng tử. Ngoài ra, vẽ đồ thị để khảo sát giá trị được lượng tử hóa theo giá trị đầu vào (tức là, giá trị được lượng tử hóa là hàm của giá trị đầu vào) ứng với các trường hợp 16, 64 và 128 mức lượng tử. Chương trình Matlab thực hiện bài toán này được cho ở file **NVD3_sim38.m** trong Phụ lục 3A. Kết quả chạy chương trình này được cho ở hình 3.22.

Thấy rõ từ hình 3.22 khi số mức lượng tử càng lớn thì sai số lượng tử càng nhỏ, như được mong đợi. Ngoài ra cũng cần lưu ý rằng, khi số mức lượng tử càng lớn, thì quan hệ giữa tín hiệu lối vào và các giá trị được lượng tử hoá càng tiến tới một đường thẳng có độ dốc bằng 1 đi qua gốc toạ độ; nghĩa là, tín hiệu vào/ra của bộ lượng tử hóa trở nên gần như bằng nhau. Ngược lại, khi số mức lượng tử hoá nhỏ (chẳng hạn 16 mức) thì quan hệ này là khác nhau xa.



Hình 3.22: Sai số lượng tử trong sơ đồ PCM lượng tử hóa đều với 16; 64; 128 mức lượng tử

PCM với lượng tử hoá không đều

Trong trường hợp này, trước hết cho tín hiệu đầu vào qua một phần tử phi tuyến để làm giảm dải động, sau đó đưa vào một hệ thống PCM lượng tử hoá đều. Tại đầu thu, tín hiệu lối ra được cho qua bộ nghịch đảo của phần tử phi tuyến đã được sử dụng ở phần phát. Nhờ tính đối nghịch (hàm ngược của nhau) mà ảnh hưởng tổng thể là tương đương với một hệ thống PCM có khoảng cách không đều giữa các mức lượng tử. Nói chung, để truyền tín hiệu tiếng nói, thì các đặc tính phi tuyến được dùng là các đặc tính phi tuyến hoặc luật A hoặc luật μ:

Đặc tính phi tuyến luật μ được định nghĩa bởi quan hệ:

$$y = g(x) = \frac{\log\left(1 + \mu.|x|\right)}{\log\left(1+\mu\right)} sign(x) \qquad (4.4.4)$$

Trong đó x là tín hiệu lối vào chuẩn hoá ($|x| \leq 1$), μ là tham số, đặc tính phi tuyến luật μ ở dạng tiêu chuẩn $\mu = 255$.

Nghịch đảo của đặc tính phi tuyến luật μ được cho bởi

$$x = \frac{\left(1+\mu\right)^{|y|} - 1}{\mu} sign(y) \qquad (4.4.5)$$

Thực thi hóa đặc tính phi tuyến luật μ và nghịch đảo của nó bằng Matlab được cho ở file **NVD3_mulaw.m** và **NVD3_invmulaw.m** trong Phụ lục 3A. Hàm **NVD3_mula_pcm.m** là tương đương với hàm **NVD3_u_pcm.m** khi dùng sơ đồ PCM luật μ. Chương trình Matlab này được cho ở file **NVD3_mula_pcm.m** trong Phụ lục 3A.

Chương 4

MÔ HÌNH MÔ PHỎNG THÔNG THẤP CHO TÍN HIỆU VÀ HỆ THỐNG THÔNG DẢI

4.1. Mở đầu

Thấy rõ từ các chương trước, cần rất nhiều mẫu để tạo ra một tín hiệu RF thông dải, do đó tín hiệu thông dải RF thường không được sử dụng trong mô phỏng ở mức dạng sóng. Tín hiệu RF là tín hiệu thông dải trong đó tần số sóng mang f_c thường lớn hơn rất nhiều độ rộng băng tần B của nó. Biểu diễn thông thấp cho tín hiệu thông dải sẽ nhận được mô phỏng nhanh hơn nhiều, giảm đáng kể việc lưu trữ số liệu cũng như các yêu cầu xử lý số liệu. Việc sử dụng mô hình thông thấp cho tín hiệu và hệ thống trong mô phỏng là trọng tâm của chương này.

Ngoài ra, nếu ta chưa quen với Matlab và muốn tìm hiểu phổ và công cụ phân tích phổ cũng như khảo sát các tín hiệu và hệ thống tuyến tính bất biến, thì thông qua các bài tập ở dạng chương trình mô phỏng trên Matlab ở mức cơ bản dễ hiểu hơn được cho ở Phụ lục 4A, các chương trình Matlab này được thiết kế rất đơn giản cho các bài tập đơn giản.

Muốn vậy, chương này sẽ nghiên cứu lý thuyết tín hiệu và hệ thống trên cơ sở đường bao phức thông thấp của các tín hiệu và hệ thống thông dải. Động cơ của việc sử dụng đường bao phức đối với các tín hiệu và hệ thống thông dải trong mô phỏng là hiệu quả tính toán. Thông qua việc dùng đường bao phức sẽ giảm đáng kể số lượng mẫu để biểu diễn tín hiệu thông dải. Sử dụng các kỹ thuật này sẽ trực tiếp giảm đáng kể thời gian mô phỏng. Vì vậy, quá trình nghiên cứu sau này ta cố gắng mô hình hoá cho tất cả các tín hiệu và hệ thống thông dải bằng các mô hình đường bao thông thấp. Hai vấn đề cơ bản trong chương cần được lưu ý:

Vấn đề thứ nhất, triển khai các mô hình cho các tín hiệu và hệ thống trên cơ sở biểu diễn đường bao phức thông thấp của các tín hiệu thông dải và đáp ứng xung kim tương đương thông thấp của các hệ thống thông dải.

Vấn đề thứ hai, triển khai các kỹ thuật để tính toán đường bao phức thông thấp đối với đầu ra hệ thống khi biết đường bao phức thông thấp của đầu vào hệ thống và mô hình thông thấp cho hệ thống đó.

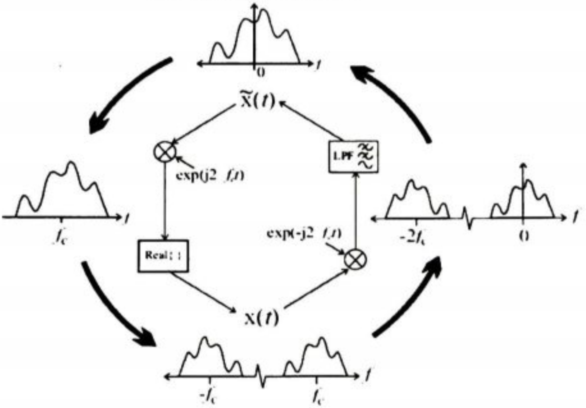
Sẽ được sáng tỏ là, mô hình đường bao phức thông thấp cho các tín hiệu thông dải được xác định theo $x_d(t)$ *và* $x_q(t)$ là các phần thực và phần ảo của đường bao phức thông thấp $\tilde{x}(t)$. Việc xử lý đường bao phức đầu vào hệ thống thông qua mô hình hệ thống được xác định bởi $h_d(t)$ và $h_q(t)$ thường bao gồm bốn phép tích chập thực. Có thể giảm được 2 lần số lượng tính toán cho vấn đề này nếu hàm truyền đạt của hệ thống thông dải có tính đối xứng liên hợp phức qua tần số tham chuẩn f_0. Việc giảm tải tính toán do $h_q(t) = 0$ đối với trường hợp đối xứng liên hợp phức. Trong nhiều trường hợp thực tế, $h_q(t) \neq 0$ nhưng ta vẫn có thể bỏ qua được khi so

sánh với $h_d(t)$ (nghĩa là $h_q(t) << h_d(t)$). Chẳng hạn, khi một bộ lọc có băng thông nhỏ hơn rất nhiều so với tần số trung tâm, thì ta có thể bỏ qua thành phần $h_q(t)$.

Trong khi tập trung nghiên cứu cho hệ thống tuyến tính bất biến, thì thực tế tồn tại nhiều hệ thống chứa các phần tử phi tuyến hoặc phụ thuộc thời gian hoặc cả hai tính chất này. Các hệ thống này phức tạp hơn rất nhiều so với hệ tuyến tính bất biến. Ví dụ minh hoạ sẽ được đề cập, nó có thể được phát triển cho các mô hình mô phỏng dựa trên việc biểu diễn đường bao phức. Nhiều hệ thống phức tạp hơn được đề cập chi tiết ở chương 13 và chương 14.

4.2. Đường bao phức thông thấp cho tín hiệu thông dải

Ta sẽ xét chi tiết biểu diễn đường bao phức thông thấp của tín hiệu ngẫu nhiên và tất định thông dải trong miền thời gian và miền tần số. Hình 4.1a minh họa quan hệ giữa các tín hiệu này trong miền thời gian và tần số.



Hình 4.1(a): Các chuyển đổi tín hiệu băng tần cơ sở - tín hiệu thông dải trong miền thời gian (vòng trong) và miền tần số (vòng ngoài)

4.2.1. Đường bao phức - miền thời gian

Một tín hiệu thông dải nói chung chẳng hạn đầu ra của bộ điều chế được viết ở dạng:

$$x(t) = A(t)\cos[2\pi f_0 t + \phi(t)] \tag{4.1}$$

trong đó $A(t)$ là biên độ (đường bao thực) của tín hiệu, $\phi(t)$ là độ lệch pha so với pha $2\pi f_0 t$. Trường hợp (4.1) biểu diễn đầu ra của một bộ điều chế, thì f_0 là tần số sóng mang và $2\pi f_0 t$ là pha tức thời của sóng mang không điều chế. Theo công thức Euler, (4.1) được viết là:

$$x(t) = \text{Re}\left\{ \underbrace{A(t)e^{j\phi(t)}}_{\tilde{x}(t)} e^{j2\pi f_0 t} \right\} \tag{4.2}$$

$$\Leftrightarrow x(t) = \text{Re}\left\{ \tilde{x}(t) e^{j2\pi f_0 t} \right\} \tag{4.3}$$

Đường bao phức $\tilde{x}(t)$ của tín hiệu thực $x(t)$ là:

$$\tilde{x}(t) = \underbrace{\underbrace{A(t)}_{\text{Đường bao thực}} \; e^{j\phi(t)}}_{\text{Đường bao phức}} \tag{4.4}$$

Phương trình (4.4) là dạng cực hay dạng mũ của đường bao phức.

Ta cũng thường biểu diễn đường bao phức ở dạng:

$$\tilde{x}(t) = x_d(t) + jx_q(t) \tag{4.5}$$

Phần thực và phần ảo của đường bao phức được gọi là thành phần đồng pha (cùng hướng) và vuông pha của $x(t)$. Áp dụng công thức Euler cho (4.4) ta có:

$$\tilde{x}(t) = \underbrace{A(t)\cos\phi(t)}_{x_d(t)} + j\underbrace{A(t)\sin\phi(\text{t})}_{x_q(t)} \tag{4.6}$$

Theo đó, ta có các thành phần đồng pha và vuông pha là:

$$x_d(t) = A(t)\cos\phi(t) \tag{4.7}$$

$$x_q(t) = A(t)\sin\phi(t) \tag{4.8}$$

Trong các ứng dụng thực tế $A(t)$ và $\phi(t)$ là hàm thông thấp có độ rộng băng tần nhỏ hơn nhiều so với f_0. Vì vậy, $x_d(t)$ và $x_q(t)$ cũng là các tín hiệu thông thấp. Theo định nghĩa về số phức thì $A(t)$ và $\phi(t)$ quan hệ với $x_d(t)$ và $x_q(t)$ như sau:

$$A(t) = |\tilde{x}(t)| = \sqrt{x_d^2(t) + x_q^2(t)} \tag{4.9}$$

$$\phi(t) = \arctan\frac{x_q(t)}{x_d(t)} \tag{4.10}$$

Sử dụng (4.3) và (4.5), tín hiệu $x(t)$ được viết là:

$$x(t) = \text{Re}\left\{ \left[x_d(t) + jx_q(t) \right] \times \left[\cos 2\pi f_0 t + j\sin 2\pi f_0 t \right] \right\} \tag{4.11}$$

$$\Rightarrow x(t) = x_d(t)\cos(2\pi f_0 t) - x_q(t)\sin(2\pi f_0 t) \tag{4.12}$$

Lưu ý rằng, (4.12) đã được viết bằng cách áp dụng đồng nhất lượng giác:

$$\cos(a+b) = \cos(a)\cos(b) - \sin(a)\sin(b) \tag{4.13}$$

Cho (4.1) và xác định $x_d(t)$ và $x_q(t)$ dùng (4.7) và (4.8).

Mặc dù f_0 thường được chọn là tần số trung tâm của tín hiệu thông dải nhưng f_0 là bất kỳ và có thể chọn tùy ý cho thuận tiện. Tuy nhiên, sẽ được minh họa trong ví dụ 4.1, $x_d(t)$ và $x_q(t)$ phụ thuộc vào việc chọn f_0. Ví dụ 4.2 minh hoạ trường hợp biểu diễn thông thấp cho tín hiệu FM tương tự. Ví dụ 4.3, 4.4 và 4.5 minh hoạ ứng dụng cho tín hiệu số, minh họa việc triển khai các mô hình mô phỏng cho các bộ điều chế số.

Ví dụ 4.1: Xét tín hiệu thông dải

$$x(t) = A.\sin\left(2\pi f_m t\right).\cos\left(2\pi f_c t + \phi\right) \tag{4.14}$$

Trong đó f_c là tần số sóng mang và ϕ là độ lệch pha sóng mang. Giả sử $f_c >> f_m$ và $x_d(t)$ & $x_q(t)$ như được định nghĩa bởi (4.3) và (4.5). Trước tiên là chọn tần số f_0 được xác định trong (4.3). Để không cho $f_0 = f_c$ ta đặt $f_c = f_0 + f_\Delta$ dẫn đến:

$$x(t) = A.\sin\left(2\pi f_m t\right)\cos.\left(2\pi f_0 t + 2\pi f_\Delta t + \phi\right) \tag{4.15}$$

$$x(t) = \mathrm{Re}\left\{\underbrace{A.\sin\left(2\pi f_m t\right).e^{j2\pi f_\Delta t}.e^{j\phi}}_{\tilde{x}(t)}.e^{j2\pi f_0 t}\right\} \tag{4.16}$$

Bằng cách kiểm tra, đường bao phức là:

$$\tilde{x}(t) = A.\sin\left(2\pi f_m t\right).e^{j(2\pi f_\Delta t + \phi)} \tag{4.17}$$

Vì vậy:

$$\tilde{x}(t) = A.\sin\left(2\pi f_m t\right).\left[\cos\left(2\pi f_\Delta t + \phi\right) + j\sin(2\pi f_\Delta t + \phi)\right] \tag{4.18}$$

Từ đó xác định được các phần thực và phần ảo (đồng pha và vuông pha) là:

$$x_d(t) = A.\sin\left(2\pi f_m t\right).\cos\left(2\pi f_\Delta t + \phi\right) \tag{4.19}$$

$$x_q(t) = A.\sin\left(2\pi f_m t\right).\sin(2\pi f_\Delta t + \phi) \tag{4.20}$$

Lưu ý là cả $x_d(t)$ và $x_q(t)$ đều phụ thuộc vào quan hệ giữa f_0 và f_c. Thể hiện rõ nhất nếu $f_0 = f_c$ ($f_\Delta = 0$), song như đã đề cập, chọn f_0 là tùy ý. Trong các bài toán mô phỏng ta chọn f_0 sao cho giảm thiểu khối lượng tính toán.

Ví dụ 4.2: Một bộ điều chế tín hiệu FM liên tục được định nghĩa bởi biểu thức:

$$x_c(t) = A_c\cos\left(2\pi f_c t + k_f\int_{t_0}^{t} m\left(\alpha\right)d\alpha + \phi(t_0)\right) \tag{4.21}$$

Trong đó, A_c và f_c là biên độ và tần số của sóng mang không điều chế, $m(t)$ là bản tin hay tín hiệu mang tin, k_f là chỉ số điều chế, t_0 là thời điểm tham chiếu bất kỳ, và $\phi\left(t_0\right)$ là độ lệch pha tại thời điểm t_0. Giả sử, chọn thời điểm tham chiếu $t_0 = 0$ và $\phi\left(t_0\right) = 0$ thì $x_c(t)$ được biểu diễn:

$$x_c(t) = \text{Re}\left\{\underbrace{A_c.e^{jk_f\int_0^t m(\alpha)d\alpha}}_{\tilde{x}(t)}.e^{j2\pi f_c t}\right\} \tag{4.22}$$

Theo đó, đường bao phức là:

$$\tilde{x}(t) = A_c.e^{jk_f\int_0^t m(\alpha)d\alpha} \tag{4.23}$$

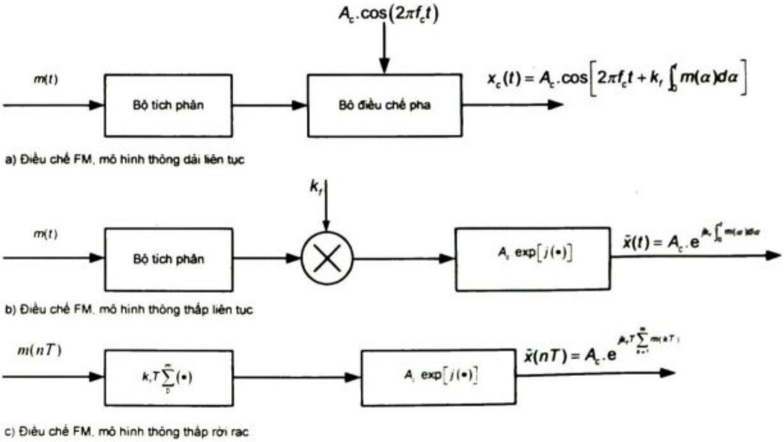
Rút ra được các thành phần đồng pha và vuông pha là:

$$x_d(t) = A_c\cos\left(k_f\int_0^t m(\alpha)d\alpha\right) \tag{4.24}$$

$$x_q(t) = A_c\sin\left(k_f\int_0^t m(\alpha)d\alpha\right) \tag{4.25}$$

Vì vậy, để trình bày $x(t)$ trong mô phỏng, chỉ cần tạo 2 tín hiệu thông thấp được cho bởi (4.24) và (4.25).

Các mô hình cho bộ điều chế FM được minh hoạ trong hình 4.1(b). Hình 4.1(b)a là mô hình thông dải liên tục theo thời gian. Mô hình tín hiệu thông thấp liên tục thời gian, trong đó đầu ra là trình bày dạng đường bao thông thấp phức của $x_c(t)$ được cho ở hình 4.1(b)b. Tương đương rời rạc được minh hoạ trong hình 4.1(b)c, với chu kỳ lấy mẫu là T và n đánh chỉ số cho các mẫu. Lưu ý rằng trong mô hình thời gian rời rạc, bộ tích phân được mô hình hoá là một tích luỹ (bộ lấy tổng). Điều này tương đương với tích phân chữ nhật trong đó diện tích được tích luỹ trong khe thời gian thứ k là $Tm(kT)$. Các mô hình bộ tích phân khác sẽ được đề cập ở chương sau.

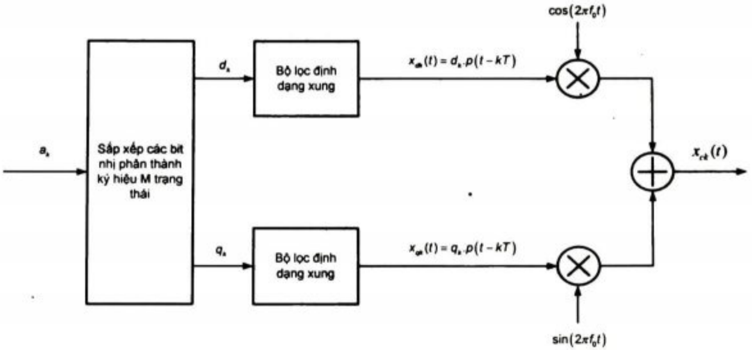


Hình 4.1(b): Các mô hình cho các quá trình điều chế FM

Xét một vài ví dụ điều chế số. Muốn vậy, cần có mô hình mô phỏng cho bộ điều chế. Mô hình cơ bản được minh họa ở hình 4.2 (Lưu ý: hình 4.2 minh họa mô hình thông dải cho đầu ra là tín hiệu thông dải $x_{ck}(x)$. Mô hình thông thấp được dùng để mô phỏng có các đầu ra $x_{dk}(t)$ và $x_{qk}(t)$ xác định đường bao phức thông thấp của $x_{ck}(t)$). Cho rằng, a_k là luồng số liệu nhị phân. Ngoài ra, tín hiệu M trạng thái, b bit mang tin được nhóm lại với nhau để tạo ra một ký hiệu số liệu do đó tín hiệu phát trong khoảng tín hiệu thứ k có thể mang nhiều hơn 1 bit thông tin. Một cách điển hình $M = 2^b$, nhưng đây không phải là giả định bắt buộc. Việc sắp xếp ký hiệu nhị phân vào M trạng thái được chỉ ra trong hình 4.2 thực hiện chức năng ghép nhóm b bit để tạo ra ký hiệu M trạng thái. Đầu ra của bộ sắp xếp là các thành phần đồng pha và vuông pha của ký hiệu thứ k. Nó được ký hiệu là d_k và q_k (ký hiệu thứ k là giá trị phức, $s_k = d_k + jq_k$).

Các ký hiệu d_k và q_k có thể được xem là hàm xung kim có các trọng số được xác định bởi sắp xếp bit vào ký hiệu. Đáp ứng xung kim của bộ lọc định dạng xung được ký hiệu là $p(t)$ do đó các tín *hiệu đồng pha và vuông pha* trong khoảng thời gian tín hiệu thứ k, $kT < t < (k+1)T$, là $x_{dk}(t)$ và $x_{qk}(t)$ như hình 4.2. Đầu ra của máy phát trong khoảng thời gian tín hiệu thứ k là:

$$x_{ck}(t) = x_{dk}(t)\cos 2\pi f_0 t - x_{qk}(t)\sin 2\pi f_0 t \tag{4.26}$$



Hình 4.2: Mô hình mô phỏng cho bộ điều chế số (mô hình thông dải)

Mô hình tín hiệu rời rạc thời gian tương ứng là:

$$x_{ck}(nT) = x_{dk}(nT)\cos 2\pi f_0 nT - x_{qk}(nT)\sin 2\pi f_0 nT \tag{4.27}$$

là phiên bản được mẫu của mô hình tín hiệu liên tục thời gian.

Trước khi kết thúc chủ đề này, ta xem lại một vài thuật ngữ, các *điểm tán xạ* (*Scattergrams*) và chỉ ra sự khác nhau giữa các điểm tán xạ khi được dùng trong tình huống mô phỏng, biểu đồ sao tín hiệu quen thuộc. Không gian tín hiệu được định nghĩa là một không gian

K chiều được tạo ra bởi K hàm trực giao cơ sở $\phi_i(t), i = 1, 2, ..., K$ và các tín hiệu được trình bày là các vectơ trong không gian này. Ví dụ: giả sử hệ thống truyền thông *M* trạng thái trong đó một trong số M tín hiệu được phát trong khoảng thời gian tín hiệu thứ *k*. Dưới dạng các hàm cơ sở trực giao, tín hiệu phát trong khoảng tín hiệu thứ k được biểu diễn:

$$x_c(t) = \sum_{i=1}^{K} x_{im}\phi_i(t), \quad kT < t < (k+1)T, \quad m = 1, 2, ..., M \tag{4.28}$$

Trong không gian tín hiệu, tín hiệu thứ m được thể hiện bởi vectơ:

$$\mathbf{X}_m = \begin{bmatrix} x_{1m} & x_{2m} & \dots & x_{Km} \end{bmatrix} \tag{4.29}$$

Không gian tín hiệu được lấy tỉ lệ sao cho $|\mathbf{X}_m|^2$ biểu diễn năng lượng của tín hiệu thứ m trong tập các đầu ra máy phát có thể. Đồ thị K chiều của các điểm $\mathbf{X}_m, m = 1,2,...,M$ xác định chòm sao tín hiệu.

Điểm tán xạ tương ứng với một tín hiệu là kết quả vẽ $x_q(t)$ theo $x_d(t)$ và được xác định dưới dạng đường bao phức thông thấp của một tín hiệu thông dải giá trị thực. Chiều của điểm tán xạ là một nếu $x_d(t) = 0$ hoặc $x_q(t) = 0$. Ngược lại chiều của điểm tán xạ là hai. Các ví dụ 4.3 và 4.4 minh họa điểm tán xạ và đồ thị sao tín hiệu có quan hệ chặt chẽ nhau khi *K* = 2, là trường hợp QPSK và QAM. Ta sẽ thấy trong ví dụ 4.5 quan hệ này sẽ không còn khi chiều của không gian tín hiệu vượt quá 2.

Ví dụ 4.3 (QPSK): Để tạo ra một tín hiệu QPSK, các ký hiệu số liệu a_k được tạo ra bằng cách lấy 2 ký hiệu nhị phân mỗi lần. Theo đó, mỗi ký hiệu số liệu bao gồm 1 trong các cặp nhị phân: 00, 01, 10, 11. Tín hiệu QPSK thông dải trong khoảng thời gian tín hiệu thứ *k* có dạng

$$x_{ck}(t) = A_c \cos(2\pi f_0 t + \phi_k), \quad kT < t < (k+1)T \tag{4.30}$$

Trong đó ϕ_k nhận một trong các giá trị $+\pi/4, -\pi/4, +3\pi/4, -3\pi/4$.

Đường bao phức tương ứng với $x_{ck}(t)$ được viết như sau:

$$\tilde{x}_{ck}(t) = \mathrm{Re}\{A_c.e^{j\phi_k}\} \tag{4.31}$$

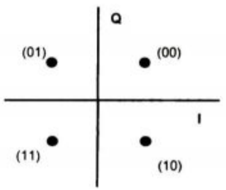
Rút ra các thành phần đồng pha và vuông pha là:

$$x_{dk}(t) = A_c \cos\phi_k \tag{4.32}$$

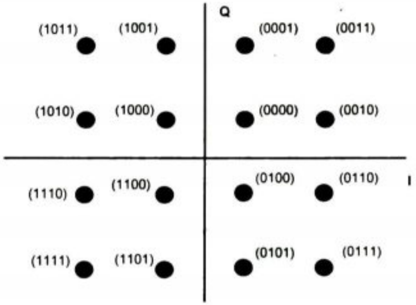
$$x_{qk}(t) = A_c \sin\phi_k \tag{4.33}$$

Lưu ý nếu $p(t)$ là một hằng số trong khoảng thời gian tín hiệu, thì cả $x_{dk}(t)$ và $x_{qk}(t)$ là hằng số trong khoảng thời gian của tín hiệu. Khi này $x_{dk}(t)$ và $x_{qk}(t)$ nhận 1 trong 2 giá trị. Nếu $x_{qk}(t)$ được vẽ là một hàm của $x_{dk}(t)$, thì điểm tán xạ được minh họa trong hình 4.3(a). Lưu ý là hình 4.3(a), khi này đã được lấy tỉ lệ một cách chính xác, cũng là đồ thị sao tín hiệu.

Mỗi một điểm trong đồ thị sao tín hiệu QPSK tương ứng với một cặp các ký hiệu nhị phân. Mặc dù cơ chế sắp xếp cặp bit nhị phân vào một ký hiệu QPSK là tùy ý nhưng việc sắp xếp được minh hoạ trong hình 4.3 thường dùng sắp xếp theo mã Gray. Trong phương pháp sắp xếp theo mã Gray, các điểm gần nhau nhất trong không gian tín hiệu chỉ khác nhau một bit nhị phân. Điều này là cần thiết, vì xác suất lỗi là hàm của khoảng cách Euclidean giữa các điểm trong không gian tín hiệu. Với cách sắp xếp trong hình 4.3, giả sử chuỗi bít là b_1b_2, trong đó b_1 là bit trọng số lớn nhất và b_2 là bit trọng số nhỏ nhất. Nếu $b_2 = 0$, thì $d_k = 1$ và nếu $b_2 = 1$, thì $d_k = -1$ do vậy bit có trọng số thấp sẽ quyết định xem khi nào ký hiệu QPSK nằm bên trái hay bên phải của mặt phẳng *I-Q*. Theo cách tương tự, nếu $b_1 = 0$, thì $q_k = 1$ và nếu $b_1 = 1$, thì $q_k = -1$ do vậy bit có trọng số lớn sẽ quyết định xem khi nào ký hiệu QPSK là ở nửa trên hay ở nửa dưới của mặt phẳng *I-Q*.



(a) Biểu đồ tán xạ của tín hiệu QPSK



(b) Biểu đồ tán xạ của tín hiệu 16-QAM

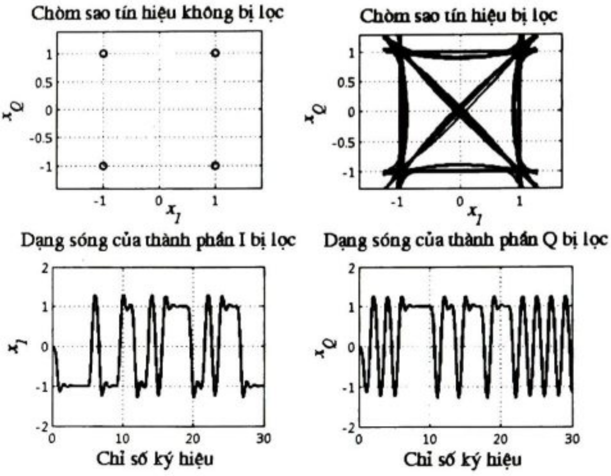
Hình 4.3: (a) Biểu đồ tán xạ của các tín hiệu QPSK và (b) 16-QAM khi p(t) là hằng số

Để khảo sát điểm tán xạ QPSK khi *p(t)* không phải hằng số và dạng sóng miền thời gian tương ứng, chương trình Matlab thực thi bài toán này được cho bởi **NVD4_qamsim.m** trong Phụ lục 4A, số mức trên trục I và trục Q được gán một cách độc lập. Với QPSK chúng đều bằng 2. Bộ lọc định dạng xung được chọn là bộ lọc Butterworth bậc sáu. Độ rộng băng của bộ lọc được dùng trong mô phỏng được ký hiệu là ***bw***, bằng tốc độ ký hiệu. Nhập các tham số tương ứng theo bảng 4.1 cho chương trình và chạy kết quả được cho ở hình 4.4.

Bảng 4.1: Các tham số đầu vào chương trình mô phỏng NVD4_qamsim.m cho QPSK

Số mức trên trục I	Số mức trên trục Q	Số ký hiệu được mô phỏng	Số mẫu trên ký hiệu	Độ rộng băng thông bộ lọc
2	2	100	20	0,1
Lưu ý: Bằng cách chọn các giá trị khác cho tập các tham số này cho phép khảo sát các ảnh hưởng của chúng lên kết quả mô phỏng. Kết quả mô phỏng cho trường hợp các giá trị này được cho ở hình 4.4				

Hàng trên là đồ thị điểm tán xạ được tạo ra bằng cách vẽ $x_q(t)$ là một hàm của $x_d(t)$. Đồ thị điểm tán xạ ở góc trên bên trái có được do $p(t)=1,\ \ 0 \le t < T_s$, trong đó T_s là chu kỳ ký hiệu (gấp đôi chu kỳ bit). Lưu ý mối quan hệ của đồ thị điểm tán xạ với đồ thị sao tín hiệu như trong hình 4.3. *Đồ thị điểm tán xạ ở góc trên bên phải được tạo ra bằng cách cho tín hiệu này qua bộ lọc Butterworth bậc sáu.* Các tín hiệu trong miền thời gian tương ứng được minh hoạ trong nửa dưới của hình 4.4, với $x_d(t)$ ở bên trái và $x_q(t)$ ở bên phải.



Hình 4.4: Các kết quả mô phỏng QPSK

Ví dụ 4.4: Sơ đồ khối máy phát QAM được cho ở hình 4.2. Bộ sắp xếp bit vào ký hiệu thực hiện sắp xếp nhóm 4 ký hiệu nhị phân đầu vào a_k thành một ký hiệu 16-QAM. Biểu đồ sao tín hiệu được minh hoạ ở hình 4.5 cùng với dãy nhị phân tương ứng cho mỗi ký hiệu 16-QAM. Cũng giống như trường hợp QPSK, việc sắp xếp các ký hiệu nhị phân thành ký hiệu 16-QAM là tùy ý nhưng việc sắp xếp này thường được xác định sao cho nhận được các kết quả mã Gray, lưu ý rằng hình 4.5 là sắp xếp mã Gray.

Trong QPSK mỗi một ký hiệu d_k và q_k nhận một trong hai giá trị là -1 hay +1. Từ hình 4.3 (b) thấy rõ cả d_k và q_k trong 16-QAM nhận một trong bốn giá trị và tồn tại trong khoảng thời gian một chu kỳ ký hiệu. Để tiện trong quá trình mô phỏng 4 giá trị được định nghĩa là: +3, +1, -1 và -3.

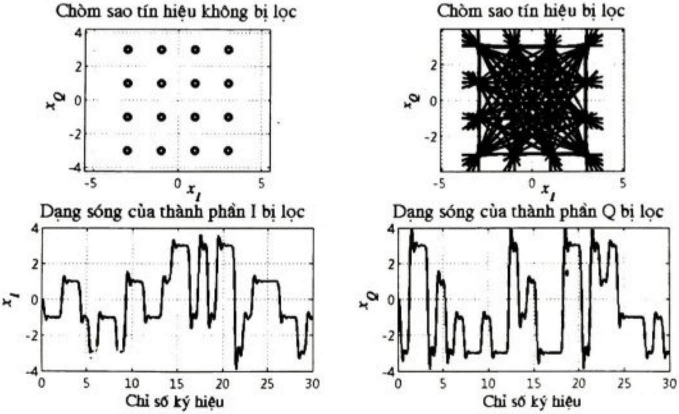
Chương trình mô phỏng cho ví dụ QPSK cũng được dùng cho ví dụ 16-QAM. Đối với 16-QAM, các mức trên trục I và trục Q đều là bốn mức. Bộ lọc định dạng xung là bộ lọc Butterworth bậc 6. Trong ví dụ 4.4, băng thông của bộ lọc gấp 2 lần tốc độ ký hiệu. Tham số đầu vào của chương trình Matlab cho trường hợp này theo bảng 4.2.

Bảng 4.2: Các tham số đầu vào chương trình mô phỏng NVD4_qamsim.m cho 16-QAM

Số mức trên trục I	Số mức trên trục Q	Số ký hiệu được mô phỏng	Số mẫu trên ký hiệu	Độ rộng băng thông bộ lọc
4	4	500	20	0,2
Lưu ý: Chọn các giá trị khác cho tập các tham số này cho phép khảo sát các ảnh hưởng của chúng lên kết quả mô phỏng. Kết quả mô phỏng trong trường hợp cụ thể này được cho ở hình 4.5.				

Nửa trên là đồ thị điểm tán xạ được tạo ra bằng cách vẽ đồ thị của $x_q(t)$ là hàm của $x_d(t)$. Đồ thị điểm tán xạ ở phía trên bên trái có được khi $p(t)=1, 0 \le t < T_s$, với T_s là chu kỳ ký hiệu (bốn lần chu kỳ bit). Lưu ý đến quan hệ giữa đồ thị điểm tán xạ và đồ thị sao trong hình 4.3 (b). Đồ thị điểm tán xạ ở phía trên bên phải được tạo ra bằng cách cho tín hiệu này qua bộ lọc Butterworth bậc 6. Tín hiệu tương ứng trong miền thời gian được minh hoạ ở hàng bên dưới của hình 4.6 với $x_d(t)$ ở góc dưới bên trái và $x_q(t)$ ở góc dưới bên phải. Có thể thấy rõ bốn giá trị (trạng thái ổn định).

Mô hình máy thu dải điều chế tín hiệu, sử dụng trong mô phỏng, tỉ số lỗi ký hiệu sẽ được ước tính. Việc ánh xạ từ tỉ số lỗi ký hiệu thành tỉ số lỗi bit có thể được thực hiện dưới dạng giải tích.



Hình 4.5: Các kết quả mô phỏng tín hiệu 16-QAM

Để tiện trình bày cho các hệ thống M-QAM, ta dùng ký hiệu A_m là các mức biên độ PAM trên các trục I và Q (hay d_k và q_k) và $b = BPS$ (Bit Per Symbol) là số bit trên một ký hiệu ứng với phương pháp điều chế cụ thể nó đánh giá hiệu quả sử dụng phổ tần của sơ đồ điều chế đó. Tạo các mức biên độ A_m cho các sơ đồ điều chế này bằng cách tạo số ngẫu nhiên phân bố đều và sắp xếp chúng thành các mức biên độ A_m trên các trục I-Q. Muốn vậy, sử dụng một bộ

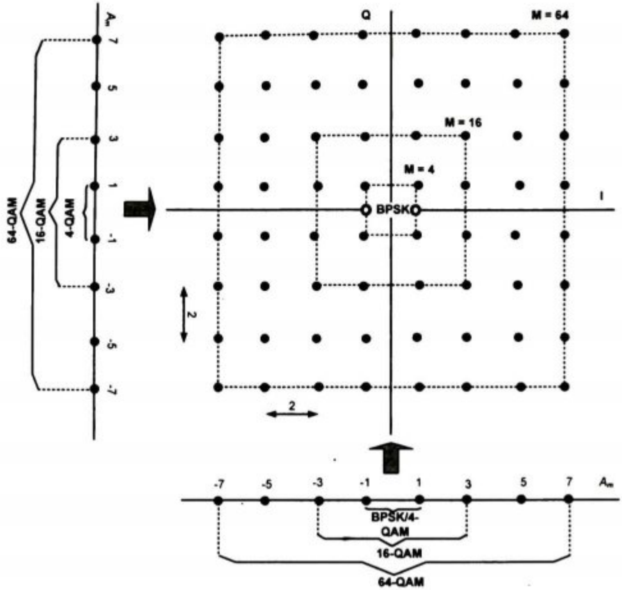
tạo số ngẫu nhiên để tạo ra số ngẫu nhiên phân bố đều trong khoảng (0,1), từ số ngẫu nhiên phân bố đều x này ánh xạ (sắp xếp) thành mức biên độ A_m tương ứng trên các trục I-Q (hay các mức d_k và q_k) cho các chế độ điều chế và giải điều chế BPSK; 4-QAM; 16-QAM; 64-QAM của M-QAM. Bộ sắp xếp (ánh xạ) thành mức biên độ A_m (PAM), luồng dữ liệu nhị phân trước hết được chuyển từ nối tiếp thành song song, sau đó được phân vào nhánh đồng pha (nhánh I) và vuông pha (nhánh Q), các tổ hợp bit của các nhánh đồng pha và vuông pha này được sắp xếp thành các mức biên độ A_m tương ứng. Quá trình này được tóm tắt trong hình 4.6.

Từ số ngẫu nhiên phân bố đều x ánh xạ thành mức biên độ A_m tương ứng với chế độ điều chế của M-QAM theo nguyên tắc được cho ở bảng 4.3.

Bảng 4.3: Sắp xếp thành mức biên độ A_m

Chế độ điều chế	Quyết định mức ra theo x (đơn vị v)	Quyết định đầu ra		Ghi chú Thời gian tồn tại giá trị A_m là T_s
		Ký hiệu tin (v)	Tổ hợp bit	
BPSK	Nếu $0 < x \le 0,5$ thì quyết định đầu ra	$A_m = -1$	0	$T_s = T_b$ BPS = 1
	Nếu $0,5 < x$ thì quyết định đầu ra	$A_m = +1$	1	
4-QAM	Nếu $0 < x \le 0,5$ thì quyết định đầu ra	$A_m = -1$	0	$T_s = 2T_b$ BPS = 2 (I hoặc Q)
	Nếu $0,5 < x$ thì quyết định đầu ra	$A_m = +1$	1	
16-QAM	Nếu $0 < x \le 0,25$ thì quyết định đầu ra	$A_m = -3$	00	$T_s = 4T_b$ BPS = 4 (I hoặc Q)
	Nếu $0,25 < x \le 0,5$ thì quyết định đầu ra	$A_m = -1$	01	
	Nếu $0,5 < x \le 0,75$ thì quyết định đầu ra	$A_m = +1$	10	
	Ngược lại thì quyết định đầu ra	$A_m = +3$	11	
64-QAM	Nếu $0 < x \le 0,125$ thì quyết định đầu ra	$A_m = -7$	000	$T_s = 6T_b$ BPS = 6 (I hoặc Q)
	Nếu $0,125 < x \le 0,25$ thì quyết định đầu ra	$A_m = -5$	001	
	Nếu $0,25 < x \le 0,375$ thì quyết định đầu ra	$A_m = -3$	010	
	Nếu $0,375 < x \le 0,5$ thì quyết định đầu ra	$A_m = -1$	011	
	Nếu $0,5 < x \le 0,625$ thì quyết định đầu ra	$A_m = +1$	100	
	Nếu $0,625 < x \le 0,75$ thì quyết định đầu ra	$A_m = +3$	101	
	Nếu $0,75 < x$ 0,875 thì quyết định đầu ra	$A_m = +5$	110	
	Nếu $0,875 < x \le 1$ thì quyết định đầu ra	$A_m = +7$	111	
Các mức điện áp A_m hay các tổ hợp bit trong cùng một chế độ điều chế có cùng xác suất. BPS là số bit trên một ký hiệu tin là tham số đánh giá hiệu năng thông lượng.				

Vì thời gian tồn tại của một ký hiệu tin $T_s = \log_2(M) \times T_b = BPS \times T_b$ nên khi mô phỏng nếu lấy n mẫu/bit thì phải lấy (n*BPS) mẫu/ký hiệu.



Hình 4.6: Minh họa quá trình sắp xếp A_m

Ví dụ 4.5 (4-FSK): Trong ví dụ 4.3 và 4.4, ta sử dụng không gian tín hiệu 2 chiều do vậy việc sắp xếp từ không gian tín hiệu vào mặt phẳng I-Q là đơn giản. Tại đây ta xét một ví dụ mà các ký hiệu nhị phân được ghép nhóm với nhau thành 2 như trong QPSK, nhưng chiều của không gian tín hiệu là 4. Do vậy, không gian tín hiệu được tạo ra bằng cách dùng 4 hàm trực giao cơ sở chứ không phải là 2 hàm cơ sở như trong QPSK. Với ví dụ này, các hàm cơ sở được chọn là các sóng hình sin có tần số khác nhau (do đó gọi là 4-FSK). Đặt hàm định dạng xung $p(t)$ là hằng số A trong khoảng thời gian tín hiệu thứ k, ta có:

$$s_m(t) = A\cos(2\pi f_m t), \quad kT < t < (k+1)T, \quad m = 1, 2, 3, 4 \tag{4.34}$$

Vì vậy, trong mỗi khoảng thời gian tín hiệu, một trong bốn hình sin được phát đi. Lưu ý rằng, mỗi $s_m(t)$ phải được trải qua khoảng thời gian là một số nguyên chu kỳ sóng mang trong khoảng thời gian tín hiệu T giây để đảm bảo:

$$\int_{(k-1)T}^{kT} s_i(t).s_j(t)dt = 0, \quad i, j = 1, 2, 3, 4, \quad i \neq j \tag{4.35}$$

sao cho các tín hiệu là trực giao.

Viết $s_m(t)$ ở dạng:

$$s_m(t) = A\cos[2\pi(f_m - f_0)t + 2\pi f_0 t],\quad kT < t < (k+1)T,\quad m = 1, 2, 3, 4 \qquad (4.36)$$

nhận được:

$$s_m(t) = \mathrm{Re}\left\{\underbrace{A.\mathrm{e}^{j2\pi(f_m - f_0)t}}_{\tilde{s}_m(t)}.\mathrm{e}^{j2\pi f_0 t}\right\},\quad kT < t < (k+1)T,\quad m = 1, 2, 3, 4 \qquad (4.37)$$

Vì vậy, đường bao phức của tín hiệu thứ *m* trong tập các tín hiệu phát là:

$$\tilde{s}_m(t) = A.\mathrm{e}^{j2\pi(f_m - f_0)t},\quad kT < t < (k+1)T,\quad m = 1, 2, 3, 4 \qquad (4.38)$$

Từ đó nhận được các thành phần đồng pha và vuông pha là:

$$x_{dk}(t) = A\cos[2\pi(f_m - f_0)t],\quad m = 1, 2, 3, 4 \qquad (4.39)$$

$$x_{qk}(t) = A\sin[2\pi(f_m - f_0)t],\quad m = 1, 2, 3, 4 \qquad (4.40)$$

Lưu ý: Mặc dù $p(t)$ là một hằng số, nhưng cả $x_q(t)$ và $x_d(t)$ đều thay đổi theo thời gian trong một khoảng thời gian tín hiệu.

4.2.2. Đường bao phức - miền tần số

Từ (4.3) ta có thể viết:

$$x(t) = \frac{1}{2}\tilde{x}(t)\mathrm{e}^{j2\pi f_0 t} + \frac{1}{2}\tilde{x}^*(t)\mathrm{e}^{-j2\pi f_0 t} \qquad (4.41)$$

Nhân với $\mathrm{e}^{-j2\pi f_0 t}$ ta có:

$$2x(t)\mathrm{e}^{-j2\pi f_0 t} = \tilde{x}(t) + \tilde{x}^*(t)\mathrm{e}^{-j4\pi f_0 t} \qquad (4.42)$$

$$\Rightarrow \tilde{x}(t) = 2x(t)\mathrm{e}^{-j2\pi f_0 t} - \tilde{x}^*(t)\mathrm{e}^{-j4\pi f_0 t} \qquad (4.43)$$

Như đã thấy trong phần trước, $\tilde{x}(t)$ là một tín hiệu thông thấp (tín hiệu có phổ khác không chỉ trong vùng lân cận $f = 0$). Lấy phần thông thấp của (4.43) ta có:

$$\tilde{x}(t) = Lp\left\{2x(t)\mathrm{e}^{-j2\pi f_0 t} - \tilde{x}^*(t)\mathrm{e}^{-j4\pi f_0 t}\right\} \qquad (4.44)$$

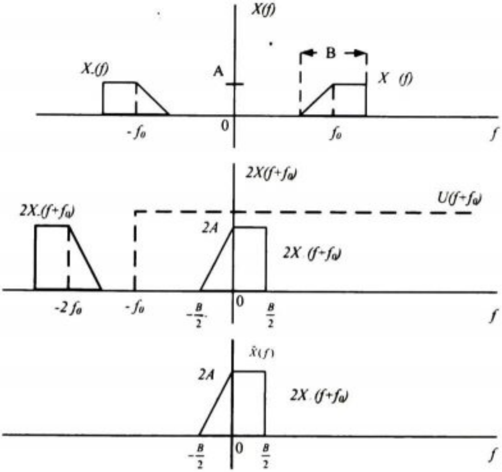
Trong đó $Lp\{.\}$ ký hiệu phần thông thấp của đối số.

Biến đổi Fourier của (4.43) ta được:

$$\tilde{X}(f) = 2X(f + f_0) - \tilde{X}^*(f + 2f_0) \qquad (4.45)$$

Như minh hoạ trong hình 4.7, $X(f)$ chỉ khác không tại các vùng tần số lân cận của $f = \pm f_0$, với $X_+(f)$ biểu diễn cho phần phổ dương của $X(f)$ và $X_-(f)$ biểu diễn cho phần phổ âm của $X(f)$. Do vậy, $\tilde{X}^*(f + 2f_0)$ khác không trong khoảng lân cận của $f = -f_0$ ($X_-(f)$

dịch trái đi một đoạn là $2f_0$) và $f = -3f_0$ ($X_-(f)$ *dịch sang trái một đoạn là* $2f_0$). Các thành phần này không tạo nên được $\tilde{X}(f)$, hay tương ứng là $\tilde{x}(t)$, vì $\tilde{X}(f)$ khác không chỉ trong lân cận của $f = 0$. Cách để $X(f+f_0)$ tạo nên $\tilde{x}(t)$ được minh hoạ trong hình 4.7. Lưu ý là $X(f+f_0)$ chỉ khác không tại các lân cận của $f = -2f_0$ ($X_-(f)$ dịch sang trái 1 đoạn là f_0) và $f = 0$ ($X_+(f)$ dịch trái đi một đoạn là f_0). Theo đó chỉ có $X_+(f+f_0)$ tạo nên $\tilde{x}(t)$ và:



Hình 4.7: Rút ra $\tilde{X}(f)$ *từ* $X(f)$

$$\tilde{X}(f) = 2X_+(f+f_0) \tag{4.46}$$

Tương đương với:

$$\tilde{x}(t) = 2Lp\left\{x(t)e^{-j2\pi f_0 t}\right\} \tag{4.47}$$

Như minh hoạ trong hình 4.7, lọc thông thấp bằng bộ lọc có hàm truyền đạt $H(f) = U(f+f_0)$. Do vậy (4.46) được biểu diễn như sau:

$$\tilde{X}(f) = 2X(f+f_0).U(f+f_0) \tag{4.48}$$

4.2.3. Rút $X_d(f)$ và $X_q(f)$ từ $\tilde{X}(f)$

Có thể dễ dàng rút ra biểu thức $X_d(f)$ và $X_q(f)$. Biến đổi Fourier của (4.5) ta được:

$$\tilde{X}(f) = X_d(f) + jX_q(f) \tag{4.49}$$

Thay *f* bằng *-f* ta có:

$$\tilde{X}(-f) = X_d(-f) + jX_q(-f) \tag{4.50}$$

Vì $x_d(t)$ và $x_q(t)$ là thực, nên $X_d(-f) = X_d^*(-f)$ và $X_q(-f) = X_q^*(-f)$ vì vậy (4.50) được viết là:

$$\tilde{X}(-f) = X_d^*(f) + jX_q^*(f) \tag{4.51}$$

Lấy liên hợp phức biểu thức (4.51) ta có:

$$\tilde{X}^*(-f) = X_d(f) - jX_q(f) \tag{4.52}$$

Cộng (4.49) và (4.52) ta có:

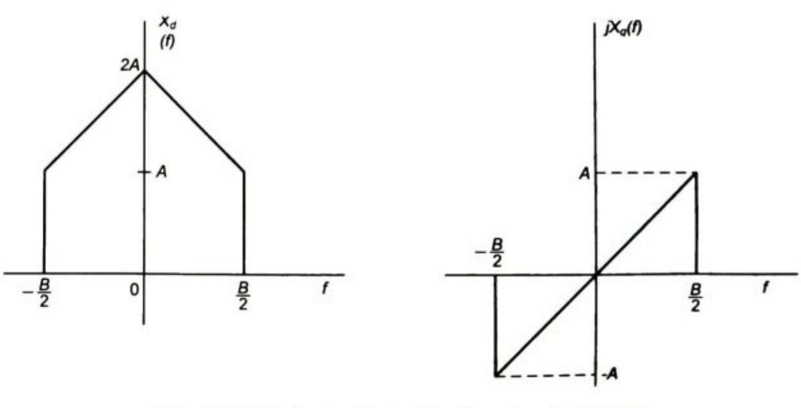
$$X_d(f) = \frac{1}{2}\left(\tilde{X}(f) + \tilde{X}^*(-f)\right) \tag{4.53}$$

Nhân (4.52) với -1 và cộng kết quả này với (4.49) thu được:

$$X_q(f) = \frac{1}{2j}\left(\tilde{X}(f) - \tilde{X}^*(-f)\right) \tag{4.54}$$

Biến đổi Fourier ngược của $X_d(f)$ và $X_q(f)$ được định nghĩa bởi (4.53) và (4.54) cho ta $x_d(t)$ và $x_q(t)$ tương ứng.

Phổ $X_d(f)$ và $X_q(f)$ tương ứng với đường bao phức thông thấp $\tilde{X}(f)$ được rút ra từ hình 4.7 được minh hoạ ở hình 4.8. Các phổ này được xác định bởi các công thức (4.53) và (4.54). Lưu ý rằng vẽ biểu đồ thành phần vuông pha thuận hơn vẽ $jX_q(f)$. Và lưu ý rằng, $X_d(f)$ thực và hàm chẵn, trong khi $X_q(f)$ là ảo và hàm lẻ.



Hình 4.8: Phổ của các thành phần đồng pha và vuông pha

Từ hình 4.7, cho thấy phổ $X(f)$ của tín hiệu thông dải giá trị thực $x(t)$ là không đối xứng qua f_0. Kết quả là, các mẫu của đường bao phức thông thấp $\tilde{x}(t)$ là các giá trị phức. Cả phần thực và phần ảo của $\tilde{x}(t)$ là $x_d(t)$ và $x_q(t)$ có băng thông là $B/2$, bằng một nửa băng thông của tín hiệu thông dải thực $x(t)$. Như đã rõ ở chương trước, cả $x_d(t)$ và $x_q(t)$ phải được lấy mẫu ở tần số lớn hơn $2(B/2) = B$ mẫu/giây. Kết quả của quá trình lấy mẫu sẽ cho ta ít nhất là *2B* mẫu trên một giây. Ngược lại, nếu $X(f)$ đối xứng liên hợp phức qua tần số f_0, thì $\tilde{X}(f)$ đối xứng liên hợp phức qua tần số $f = 0$. Khi này, $\tilde{x}(t)$ sẽ là thực (vì $x_q(t) = 0$) và việc lấy mẫu thành phần vuông pha là không cần thiết.

4.2.4. Năng lượng và công suất

Từ lý thuyết hệ thống tuyến tính, thấy rõ định lý Parseval cho biết biến đổi Fourier bảo tồn năng lượng và công suất. Đáng tiếc là, năng lượng (hay công suất) trong đường bao phức không bằng với năng lượng (hay công suất) tương ứng với tín hiệu thông dải. Sử dụng (4.41) ta có:

$$|x(t)|^2 = \frac{1}{4}\left|\tilde{x}(t)e^{j2\pi f_0 t} + \tilde{x}^*(t)e^{-j2\pi f_0 t}\right|^2 \tag{4.55}$$

đối với công suất tức thời trong $x(t)$. Biểu thức trên cho ta:

$$|x(t)|^2 = \frac{1}{4}\left[\tilde{x}(t)e^{j2\pi f_0 t} + \tilde{x}^*(t)e^{-j2\pi f_0 t}\right]\times\left[\tilde{x}^*(t).e^{-j2\pi f_0 t} + \tilde{x}(t).e^{j2\pi f_0 t}\right] \tag{4.56}$$

Khai triển phép nhân trên:

$$|x(t)|^2 = \frac{1}{4}\left[|\tilde{x}(t)|^2 + \left[\tilde{x}(t)\right]^2 e^{j4\pi f_0 t} + \left[\tilde{x}^*(t)\right]^2 e^{-j4\pi f_0 t} + |\tilde{x}(t)|^2\right] \tag{4.57}$$

Vì $\left[\tilde{x}(t)\right]^2 e^{j4\pi f_0 t}$ và $\left[\tilde{x}^*(t)\right]^2 e^{-j4\pi f_0 t}$ biểu diễn các tín hiệu thông dải, vì vậy có trung bình bằng 0, lấy trung bình ta có:

$$E\left\{|x(t)|^2\right\} = \frac{1}{2}E\left\{|\tilde{x}(t)|^2\right\} \tag{4.58}$$

Theo định nghĩa, công suất trung bình trong tín hiệu thông dải giá trị thực $x(t)$ là:

$$\varepsilon_x = E\left\{|x(t)|^2\right\} \tag{4.59}$$

Đôi khi được hiểu là công suất thực. Công suất trong đường bao phức thông thấp $\tilde{x}(t)$, mà đôi khi được hiểu là công suất phức là:

$$\varepsilon_{\tilde{x}} = E\left\{|\tilde{x}(t)|^2\right\} \tag{4.60}$$

Thế (4.59) và (4.60) vào (4.58) nhận được:

$$\varepsilon_{\tilde{x}} = 2\varepsilon_x \tag{4.61}$$

Vì vậy, công suất trong đường bao phức của tín hiệu lớn gấp 2 lần công suất trong tín hiệu thông dải giá trị thực mà từ đó nó được rút ra. Tuy nhiên, sẽ thấy ngay trong phần sau, nhận được kết quả tương tự đối với tín hiệu ngẫu nhiên và nhiễu. Vì vậy, một số đại lượng quan trọng, tỉ số tín hiệu trên tạp âm, được duy trì khi các tín hiệu thông dải giá trị thực được biểu diễn bởi đường bao phức thông thấp tương ứng của nó.

4.2.5. Mô hình cầu phương cho tín hiệu thông dải ngẫu nhiên

Việc biểu diễn tín hiệu trong miền tần số thông qua sử dụng biến đổi Fourier hàm ý rằng, tín hiệu này là tín hiệu năng lượng tất định. Tín hiệu ngẫu nhiên thông dải cũng có biểu diễn thông thấp theo thành phần đồng pha và vuông pha. Toán học cơ bản để tạo mô hình tín hiệu thông thấp cho tín hiệu thông dải ngẫu nhiên hơi khác. Ta nhắc lại lý thuyết căn bản.

Ví dụ, xét quá trình ngẫu nhiên băng hẹp được định nghĩa bởi phương trình:

$$n(t) = n_d(t)\cos(2\pi f_0 t + \theta) - n_q(t)\sin(2\pi f_0 t + \theta) \tag{4.62}$$

Trong đó θ là góc pha bất kỳ phân bố đều trong khoảng $[\pi, \pi)$. Quá trình trong (4.62) được viết như sau:

$$n(t) = R(t)\cos\left[2\pi f_0 t + \phi(t) + \theta\right] \tag{4.63}$$

$$n(t) = \mathrm{Re}\left\{\underbrace{R(t)e^{j\phi(t)}}_{\tilde{n}(t)} e^{j(2\pi f_0 t + \theta)}\right\} \tag{4.64}$$

Đường bao phức tương ứng với $n(t)$ được định nghĩa là:

$$\tilde{n}(t) = R(t)e^{j\phi(t)} \tag{4.65}$$

Trong tọa độ vuông góc:

$$\tilde{n}(t) = n_d(t) + jn_q(t) \tag{4.66}$$

Đường bao thực *R(t)* và góc pha $\phi(t)$ là:

$$R(t) = |\tilde{n}(t)| = \sqrt{n_d^2(t) + n_q^2(t)} \tag{4.67}$$

$$\phi(t) = \arctan\frac{n_q(t)}{n_d(t)} \tag{4.68}$$

Giả sử ta biết mật độ phổ công suất (PSD) của $n(t)$. Vấn đề là xác định PSD của $n_d(t)$, $n_q(t)$ và $\tilde{n}(t)$.

Việc xác định mô hình *cầu phương* cho quá trình ngẫu nhiên thông dải thường có trong lý thuyết truyền tin cơ bản. Ở đây chỉ tóm tắt các kết quả quan trọng nhất, chúng rất hữu hiệu cho các chương sau:

• **Trung bình**: Vì *n(t)* là một quá trình thông dải nên nó có trung bình bằng 0. Theo đó, kỳ vọng vế phải của (4.62) là trung bình 0 dẫn đến $n_d(t)$, $n_q(t)$ là trung bình không. Vì vậy:

$$E\{n(t)\} = E\{n_d(t)\} = E\{n_q(t)\} = 0 \tag{4.69}$$

Trong đó $E\{.\}$ là ký hiệu của phép lấy kỳ vọng.

• **Phương sai:** Cũng thấy rằng $n_d(t)$, $n_q(t)$ có cùng phương sai (hay công suất, vì quá trình này có trung bình không) và công suất này bằng tổng công suất trong quá trình thông dải. Nói cách khác:

$$E\{n^2(t)\} = E\{n_d^2(t)\} = E\{n_q^2(t)\} = N \tag{4.70}$$

Trong đó N là tổng công suất trong quá trình ngẫu nhiên thông dải cơ bản.

• **PSD của $n_d(t)$ và $n_q(t)$**: Mật độ phổ công suất PSD của $n_d(t)$ và $n_q(t)$ là bằng nhau và được xác định từ PSD của *n(t)* được ký hiệu là $S_n(f)$, được cho bởi biểu thức:

$$S_{n_d}(f) = S_{n_q}(f) = Lp[S_n(f-f_0) + S_n(f+f_0)] \tag{4.71}$$

• **Tự tương quan của $n_d(t)$ và $n_q(t)$**: Theo định lý Weiner-Khintchine thì:

$$R_{n_d}(\tau) \leftrightarrow S_{n_d}(f) \tag{4.72}$$

$$R_{n_q}(\tau) \leftrightarrow S_{n_q}(f) \tag{4.73}$$

Trong đó $R_{n_d}(\tau)$ và $R_{n_q}(\tau)$ là hàm tự tương quan của $n_d(t)$ và $n_q(t)$, ký hiệu $\leftrightarrow$ là cặp biến đổi Fourier.

• **PSD chéo**: Mật độ phổ công suất PSD chéo của $n_d(t)$ và $n_q(t)$ được cho bởi:

$$S_{n_d n_q}(f) = jLp[S_n(f-f_0) + S_n(f+f_0)] \tag{4.74}$$

Lưu ý PSD chéo là ảo.

• **Tương quan chéo của $n_d(t)$ và $n_q(t)$**: Ta lại dùng định lý Weiner-Khintchine để định nghĩa tương quan chéo của $n_d(t)$ và $n_q(t)$. Kết quả là:

$$R_{n_d n_q}(\tau) \leftrightarrow S_{n_d n_q}(f) \tag{4.75}$$

Trong đó $R_{n_d n_q}(\tau)$ là tương quan chéo của $n_d(t)$ và $n_q(t)$. Lưu ý rằng, vì $S_{n_d n_q}(f)$ là thuần ảo nên $R_{n_d n_q}(\tau)$ là hàm lẻ. Với các quá trình có băng tần hạn chế, thì tương quan chéo cũng như tự tương quan phải liên tục. Do vậy, với các quá trình hạn băng, thì $n_d(t)$ và $n_q(t)$ là không tương quan. Tuy nhiên, $n_d(t)$ và $n_q(t+\tau)$ có thể tương quan với nhau khi $\tau \neq 0$.

• **Trung bình của đường bao phức $\tilde{n}(t)$**: Trung bình của $\tilde{n}(t)$ là:

$$E\{\tilde{n}(t)\} = E\{n_d(t) + jn_q(t)\} \tag{4.76}$$

Vì kỳ vọng của tổng bằng tổng các kỳ vọng:

$$E\{\tilde{n}(t)\} = E\{n_d(t)\} + jE\{n_q(t)\} = 0 + j0 = 0 \tag{4.77}$$

• **Phương sai của đường bao phức $\tilde{n}(t)$**: Công suất trong $\tilde{n}(t)$ là:

$$P_{\tilde{n}} = E\left\{\left|\tilde{n}(t)\right|^2\right\} = E\left\{\left[n_d(t) + jn_q(t)\right] \times \left[n_d^*(t) - jn_q^*(t)\right]\right\} \tag{4.78}$$

Vì $\tilde{n}(t)$ có trung bình không nên $E\left\{\left|\tilde{n}(t)\right|^2\right\}$ cũng là phương sai, khai triển (4.78) ta có:

$$P_{\tilde{n}} = E\left\{\left|n_d(t)\right|^2\right\} + E\left\{\left|n_q(t)\right|^2\right\} + jE\left\{n_d^*(t)n_q(t)\right\} - jE\left\{n_d(t)n_q^*(t)\right\} \tag{4.79}$$

Vì $n_d(t)$ và $n_q(t)$ là không tương quan khi các quá trình hạn băng:

$$E\left\{n_d^*(t)n_q(t)\right\} = E\left\{n_d^*(t)\right\}E\left\{n_q(t)\right\} = 0 \tag{4.80}$$

Và tương tự như vậy:

$$E\left\{n_d(t)n_q^*(t)\right\} = E\left\{n_d(t)\right\}E\left\{n_q^*(t)\right\} = 0 \tag{4.81}$$

Vì vậy:

$$P_{\tilde{n}} = E\left\{\left|n_d(t)\right|^2\right\} + E\left\{\left|n_q(t)\right|^2\right\} = P_{n_d} + P_{n_q} \tag{4.82}$$

Trong đó P_{n_d} và P_{n_q} biểu diễn công suất trong $n_d(t)$ và $n_q(t)$. Từ công thức (4.70) ta có:

$$P_{\tilde{n}} = 2E\left\{n^2(t)\right\} = 2N \tag{4.83}$$

Nghĩa là công suất trong trình bày đường bao phức thông thấp của tín hiệu thông dải là gấp đôi công suất của tín hiệu thông dải giá trị thực mà từ tín hiệu thông dải thực này rút ra đường bao phức thông thấp.

• **PSD của đường bao phức $\tilde{n}(t)$:** Từ (4.71) cho thấy PSD của $n_d(t)$ và $n_q(t)$ là bằng nhau. Vì vậy:

$$S_{\tilde{n}}(f) = S_{n_d}(f) + S_{n_q}(f) = 2S_{n_d}(f) \tag{4.84}$$

Ta sẽ minh họa tại sao quan hệ trên lại quan trọng trong môi trường mô phỏng.

4.2.6. Tỉ số tín hiệu trên tạp âm

Thấy rõ từ lý thuyết truyền thông cơ bản, tỉ số tín hiệu trên tạp âm SNR ở đầu vào của máy thu là tham số chính để xác định hiệu năng của hệ thống. Tại đầu vào máy thu, tín hiệu và tạp âm đều là thông dải.

Giả sử tín hiệu $x(t)$ và tạp âm $n(t)$ cộng với nhau, thì tín hiệu đầu vào của máy thu là:

$$z(t) = x(t) + n(t) \tag{4.85}$$

Tỉ số tín hiệu trên tạp âm dưới dạng các tín hiệu thông dải thực được định nghĩa là:

$$(SNR)_{\text{thông dải}} = \frac{E\left\{x^2(t)\right\}}{E\left\{n^2(t)\right\}} \tag{4.86}$$

Theo (4.58) và (4.83) thì:

$$(SNR)_{\text{thông dải}} = \frac{\frac{1}{2}E\left\{\left|\tilde{x}(t)\right|^2\right\}}{\frac{1}{2}E\left\{\left|\tilde{n}(t)\right|^2\right\}} = \frac{E\left\{\left|\tilde{x}(t)\right|^2\right\}}{E\left\{\left|\tilde{n}(t)\right|^2\right\}} = (SNR)_{\text{thông thấp}} \tag{4.87}$$

Trong đó $(SNR)_{\text{thông dải}}$ và $(SNR)_{\text{thông thấp}}$ được hiểu là tỉ số tín hiệu trên tạp âm dựa vào các tín hiệu thông dải giá trị thực và các đường bao phức thông thấp tương ứng. Kết quả rất quan trọng này cho thấy, việc biểu diễn các hàm thông dải (cả tín hiệu và tạp âm) bởi các thành phần tương đương thông thấp tương ứng là *phương pháp luận mô phỏng tiêu chuẩn, bảo tồn các tỉ số tín hiệu trên tạp âm*. Một ví dụ đơn giản để minh hoạ vấn đề quan trọng này bằng cách xét các tín hiệu cơ bản trong miền tần số.

Ví dụ 4.6: (Biến đổi SNR), giả sử rằng một tín hiệu thông dải được biểu diễn bởi:

$$\begin{aligned} z(t) &= x(t) + n(t) \\ &= x(t) + n_d(t)\cos(2\pi f_0 t + \theta) - n_q(t)\sin(2\pi f_0 t + \theta) \end{aligned} \tag{4.88}$$

Trong đó:

$$x(t) = A\cos\left(2\pi f_0 t\right) \tag{4.89}$$

Tín hiệu thông dải có thể được biểu diễn:

$$x(t) = \frac{A}{2}e^{j2\pi f_0 t} + \frac{A}{2}e^{-j2\pi f_0 t} \tag{4.90}$$

Vì vậy, PSD của $x(t)$ là:

$$S_x(f) = \frac{A^2}{4}\delta\left(f - f_0\right) + \frac{A^2}{4}\delta\left(f + f_0\right) \tag{4.91}$$

Như được minh hoạ trong hình 4.9(a). Do vậy, toàn bộ công suất trong tín hiệu thông dải thực là:

$$P_x = \int_{-\infty}^{\infty} S_x(f)df = \frac{A^2}{2} \tag{4.92}$$

Như được mong đợi từ (4.89). PSD của tạp âm giả định được minh hoạ trong hình 4.9(b). Vì vậy toàn bộ công suất tạp âm là:

$$P_n = \int_{-\infty}^{\infty} S_n(f)df = 2\left(\frac{N_0}{2}B\right) = N_0 B \tag{4.93}$$

Từ (4.92) và (4.93) ta được tỉ số tín hiệu trên tạp âm thông dải.

$$(SNR)_{\text{thông dải}} = \frac{P_x}{P_n} = \frac{A^2}{2N_0B} \tag{4.94}$$

Tại đây ta quan tâm các tương đương thông thấp của $x(t)$ và $n(t)$. Phương trình (4.89) có thể viết lại như sau:

$$x(t) = \text{Re}\left\{A.e^{j2\pi f_0 t}\right\} \tag{4.95}$$

Từ đó, đường bao phức là:

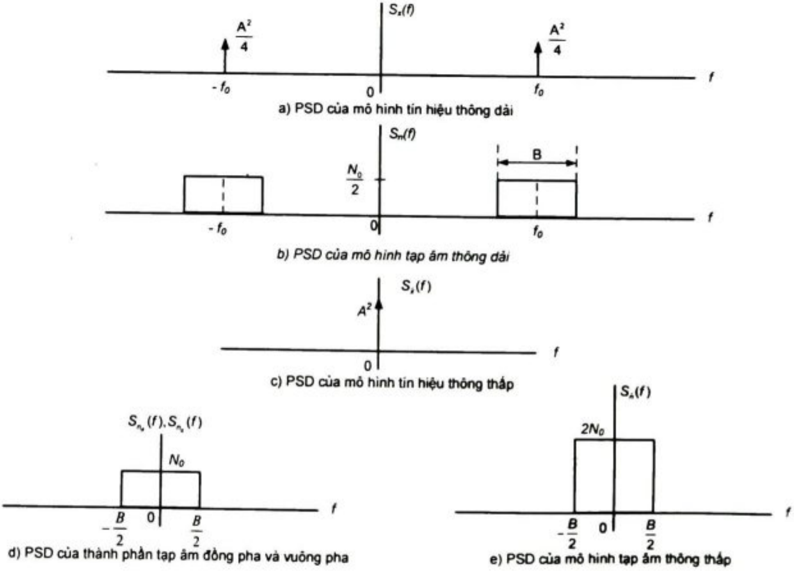
$$\tilde{x}(t) = A \tag{4.96}$$

Công suất trong đường bao phức là:

$$P_{\tilde{x}} = E\left\{\left|\tilde{x}(t)\right|^2\right\} = A^2 \tag{4.97}$$

Điều này cho ta PSD:

$$P_{\tilde{x}}(f) = A^2\delta(f) \tag{4.98}$$



Hình 4.9: Tính toán tỉ số tín hiệu trên tạp âm SNR

Như được minh họa ở hình 4.9(c). PSD của $n_d(t)$ và $n_q(t)$ được cho bởi (4.71) và được minh họa trong hình 4.9(d). Từ (4.84) PSD của $\tilde{n}(t)$ tìm được bằng cách nhân PSD trong hình 4.9(d) với 2. Kết quả được minh họa trong hình 4.9(e). Từ hình 4.9(e) công suất trong biểu diễn thông thấp phức của tạp âm là:

$$P_{\tilde{n}} = \int_{-\infty}^{\infty} S_{\tilde{n}}(f)df = 2N_0B \tag{4.99}$$

Kết hợp (4.97) và (4.99) nhận được:

$$(SNR)_{\text{thông thấp}} = \frac{P_{\tilde{s}}}{P_{\tilde{n}}} = \frac{A^2}{2N_0B} \tag{4.100}$$

So sánh (4.94) và (4.100) thấy rõ, tỉ số tín hiệu trên tạp âm được bảo tồn khi các tín hiệu thông dải giá trị thực được thay bởi các tương đương thông thấp phức của chúng.

4.3. Hệ thống thông dải tuyến tính

Ta chuyển từ quan tâm tín hiệu sang hệ thống. Bài toán cơ bản là xác định quan hệ vào/ra trong miền thời gian của hệ thống tuyến tính với giả thiết tín hiệu vào hệ thống và đáp ứng xung kim của hệ thống đều là tín hiệu thông dải được biểu diễn dưới dạng các đường bao phức thông thấp. Kết quả này sẽ cho ta phương pháp luận để triển khai các mô phỏng cho các hệ thống tuyến tính ở mức dạng sóng dựa vào các mô hình thông thấp.

4.3.1 Hệ thống tuyến tính bất biến

Giả sử hệ thống là tuyến tính, ta sử dụng *tích chập* để xác định đầu ra $y(t)$ khi đã biết đầu vào $x(t)$. Trường hợp bất biến, tích chập có dạng đơn giản:

$$y(t) = \int_{-\infty}^{\infty} x(\lambda)h(t-\lambda)d\lambda \triangleq x(t)\otimes h(t) \tag{4.101}$$

Trong đó $h(t)$ là *đáp ứng xung kim đơn vị* của hệ thống bất biến, và $\otimes$ là phép tích chập. Theo định nghĩa, đường bao phức của đầu vào hệ thống tuyến tính bất biến $\tilde{x}(t)$ và đầu ra $\tilde{y}(t)$ được định nghĩa bởi:

$$x(t) = \text{Re}\left\{\tilde{x}(t).e^{j2\pi f_0 t}\right\} \tag{4.102}$$

$$y(t) = \text{Re}\left\{\tilde{y}(t).e^{j2\pi f_0 t}\right\} \tag{4.103}$$

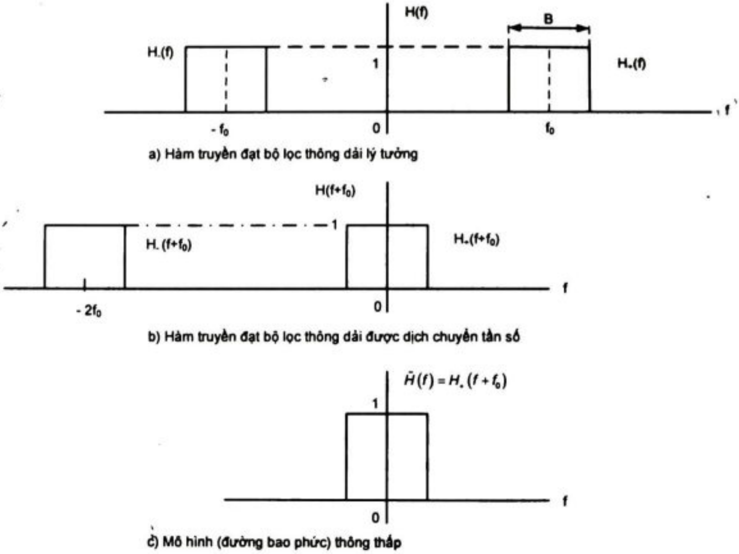
Nếu ta cần mối quan hệ giữa $\tilde{x}(t)$ và $\tilde{y}(t)$ thỏa mãn:

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{x}(\lambda)\tilde{h}(t-\lambda)d\lambda \triangleq \tilde{x}(t)\otimes \tilde{h}(t) \tag{4.104}$$

Sao cho (4.101) và (4.104) có cùng dạng, thì đáp ứng xung kim đơn vị của hệ thống thông dải $h(t)$ và đường bao phức tương ứng $\tilde{h}(t)$ phải thỏa mãn theo quan hệ:

$$h(t) = \text{Re}\left\{2\tilde{h}(t)e^{j2\pi f_0 t}\right\} \tag{4.105}$$

Lý do của hệ số 2 đó là bộ lọc thông dải độ lợi đơn vị tương ứng với việc biểu diễn đường bao phức của bộ lọc thông thấp độ lợi đơn vị.



Hình 4.10: Bộ lọc thông dải lý tưởng và mô hình thông thấp

Phương trình (4.105) dễ dàng chứng minh là đúng bằng cách chỉ ra rằng hệ số 2 trong (4.105) là do chuyển bộ lọc thông dải độ lợi đơn vị thành bộ lọc thông thấp độ lợi đơn vị vì vậy hệ số 2 duy trì độ lợi thông dải bộ lọc. Xét đặc tính bộ lọc thông dải lý tưởng được cho ở hình 4.10(a). Hàm truyền đạt của một bộ lọc lý tưởng được biểu diễn bởi:

$$H(f) = H_+(f) + H_-(f) \qquad (4.106)$$

Trong đó $H_+(f)$ và $H_-(f)$ là thành phần tần số dương và thành phần tần số âm của hàm truyền đạt bộ lọc thông dải. Thế f bằng $f + f_0$ ta có:

$$H(f + f_0) = H_+(f + f_0) + H_-(f + f_0) \qquad (4.107)$$

Như được minh hoạ trong hình 4.10(b). Rõ ràng $H_+(f + f_0)$ là hàm thông thấp, do vậy được xác định là $\tilde{H}(f)$. Vì vậy:

$$\tilde{H}(f) = H_+(f + f_0) \qquad (4.108)$$

Như được thấy ở hình 4.10(c). Lưu ý rằng (4.108) cũng có thể viết lại như sau:

$$\tilde{H}(f) = H(f + f_0).U(f + f_0) \qquad (4.109)$$

Thấy rõ, bộ lọc *thông dải* độ lợi đơn vị sắp xếp bộ lọc *thông thấp* độ lợi đơn vị, vì $\tilde{H}(f)$ được rút ra từ thành phần tần số dương của hàm truyền đạt đối với bộ lọc thông dải bằng cách chuyển dịch tần số và không cần đến tỉ số biên độ.

Cần phải hiểu rõ sự khác biệt giữa (4.48) và (4.109). Việc biểu diễn tín hiệu thông dải bởi đường bao phức thông thấp của nó dẫn đến:

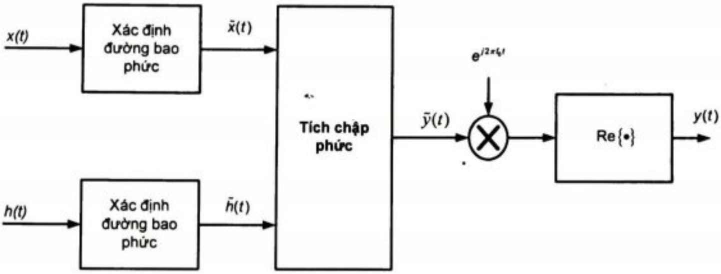
$$\tilde{X}(f) = 2X(f + f_0).U(f + f_0) \tag{4.110}$$

Trong khi đó biểu diễn các hệ thống thông dải tuyến tính bằng tương đương thông thấp dẫn đến:

$$\tilde{H}(f) = H(f + f_0).U(f + f_0) \tag{4.111}$$

Không tính đến hệ số 2, thì hai phương trình là hoàn toàn tương đương.

a) Phân tích hệ thống sử dụng các tín hiệu thông dải



b) Phân tích hệ thống sử dụng các tín hiệu đường bao phức

Hình 4.11: Kỹ thuật phân tích tín hiệu trong miền thời gian

Từ (4.101) và (4.104), ta thấy 2 kỹ thuật được dùng để tính tín hiệu ra của hệ thống tuyến tính bất biến LTIV khi biết tín hiệu thông dải đầu vào và đáp ứng xung kim trong miền thời gian. Hai kỹ thuật đó được minh hoạ trong hình 4.11. Kỹ thuật đầu tiên chỉ là tính chập $x(t)$ và $h(t)$ như được định nghĩa bởi (4.101) và được minh hoạ ở hình 4.11(a). Kỹ thuật thứ hai thực hiện theo hai bước: *(i)* xác định đường bao phức của $x(t)$ và $h(t)$, tính tích chập các đường bao phức được định nghĩa bởi (4.104); *(ii)* dùng (4.103) để xác định tín hiệu thông dải đầu ra $y(t)$. Kỹ thuật này được cho ở hình 4.11(b).

Sử dụng (4.104) ta có thể viết như sau:

$$\tilde{y}(t) = y_d(t) + jy_q(t) = \left[x_d(t) + jx_q(t)\right] \otimes \left[h_d(t) + jh_q(t)\right] \tag{4.112}$$

Vì tích chập của tổng là tổng các tích chập. Ta có:

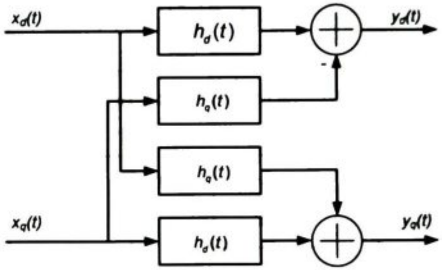
$$\begin{aligned} y_d(t) + jy_q(t) = &\left[x_d(t) \otimes h_d(t) - x_q(t) \otimes h_q(t)\right] \\ &+ j\left[x_d(t) \otimes h_q(t) + x_q(t) \otimes h_q(t)\right] \end{aligned} \tag{4.113}$$

Vì vậy, thành phần đồng pha và vuông pha của tín hiệu ra hệ thống tuyến tính được cho bởi:

$$y_d(t) = x_d(t) \otimes h_q(t) - x_q(t) \otimes h_d(t) \tag{4.114}$$

$$y_q(t) = x_d(t) \otimes h_q(t) + x_q(t) \otimes h_d(t) \tag{4.115}$$

Do vậy, tích chập của hai hàm phức tương đương với bốn *tích chập thực* giống như việc nhân 2 số phức tương đương với phép nhân 4 số thực. Quá trình để rút ra các đầu ra đồng pha và vuông pha của hệ thống thông dải tuyến tính được minh họa ở hình 4.12.



Hình 4.12: Mô hình cho hệ thống thông dải tuyến tính

Ví dụ 4.7: Trong ví dụ này ta xác định giá trị của $h_d(t)$ và $h_q(t)$ cho bộ dịch pha thông dải. Giả sử tín hiệu vào/ra hệ thống (bộ dịch pha thông dải) là:

$$x(t) = A\cos(2\pi f_0 t + \theta) \tag{4.116}$$

$$y(t) = A\cos(2\pi f_0 t + \theta + \phi) \tag{4.117}$$

Vì vậy, hệ thống làm dịch pha tín hiệu vào một lượng ϕ.

Mô hình này có thể được dùng để trình bày lỗi đồng bộ trong bộ giải điều chế. Để mô phỏng thiết bị này sử dụng các mô hình thông thấp phức, cần phải có $h_d(t)$ và $h_q(t)$.

Đường bao phức của $x(t)$ và $y(t)$ được cho bởi:

$$\tilde{x}(t) = A\,e^{j\theta} \tag{4.118}$$

$$\tilde{y}(t) = A\,e^{j(\theta+\phi)} = A\,e^{j\theta}e^{j\phi} \tag{4.119}$$

Theo đó:

$$\tilde{y}(t) = \tilde{x}(t)e^{j\phi} \tag{4.120}$$

Viết trong hệ trục tọa độ vuông góc ta có:

$$\left[y_d(t)+jy_q(t)\right]=\left[x_d(t)+jx_q(t)\right]\left[\cos\phi+j\sin\phi\right] \quad (4.121)$$

Lấy phần thực 2 vế phương trình ta có:

$$y_d(t)=x_d(t)\cos\phi-x_q(t)\sin\phi \quad (4.122)$$

Lấy phần ảo 2 vế phương trình ta có:

$$y_q(t)=x_d(t)\sin\phi+x_q(t)\cos\phi \quad (4.123)$$

Trong ví dụ này $x_d(t)$ và $x_q(t)$ lần lượt được nhân với $\cos\phi$ và $\sin\phi$. Vì vậy:

$$h_d(t)=(\cos\phi).\delta(t) \quad (4.124)$$

$$h_q(t)=(\sin\phi).\delta(t) \quad (4.125)$$

Lưu ý rằng, xuất hiện hàm delta là do hệ thống *không nhớ.*

Mô hình thông thấp dịch pha được minh hoạ ở hình 4.12 với $h_d(t)$ và $h_q(t)$ được cho bởi (4.124) và (4.125). Tất nhiên, ϕ có thể thay đổi theo thời gian.

4.3.2. Rút $h_d(t)$ và $h_q(t)$ từ $H(f)$

Để mô phỏng hệ thống sử dụng thành phần thông dải (chẳng hạn bộ lọc thông dải), ta biết hàm truyền đạt $H(f)$. Mô hình mô phỏng thông thấp cho bộ lọc được minh hoạ ở hình 4.12. Để triển khai mô hình mô phỏng cho bộ lọc dựa trên đường bao phức của $h(t)$, cần phải xác định $h_d(t)$ và $h_q(t)$ từ hàm truyền đạt của bộ lọc thông dải $H(f)$. Hai phương pháp cơ bản để tìm $h_d(t)$ và $h_q(t)$. *Phương pháp thứ nhất* là xác định $H_d(f)$ và $H_q(f)$ từ $H(f)$, sau đó biến đổi Fourier ngược để tìm $h_d(t)$ và $h_q(t)$. *Phương pháp thứ hai* là xác định $\tilde{H}(f)$ từ $H(f)$, biến đổi Fourier ngược của $\tilde{H}(f)$ để tìm $\tilde{h}(t)$, sau đó lấy phần thực và phần ảo của $h(t)$ để tìm $h_d(t)$ và $h_q(t)$.

Bước đầu tiên của 2 phương pháp là tìm $h_d(t)$ và $h_q(t)$ từ $H(f)$ để xác định $\tilde{H}(f)$. Theo định nghĩa

$$\tilde{H}(f)=H_+(f+f_0)=H_d(f)+jH_q(f) \quad (4.126)$$

Thực hiện biến đổi Fourier ngược ta được:

$$\tilde{h}(t)=h_d(t)+jh_q(t) \quad (4.127)$$

Lấy phần thực và phần ảo của $\tilde{h}(t)$ cho ta $h_d(t)$ và $h_q(t)$.

Thế f bằng $-f$ vào (4.126) ta có:

$$\tilde{H}(-f) = H_d(-f) + jH_q(-f) \tag{4.128}$$

Vì cả $h_d(t)$ và $h_q(t)$ đều là hàm thực của biến thời gian, lý thuyết biến đổi Fourier cơ bản cho ta $H_d(-f)$ là liên hợp phức của $H_d(f)$, $H_q(-f)$ là liên hợp phức của $H_q(f)$. Vì vậy, (4.128) được viết như sau:

$$\tilde{H}(-f) = H_d^*(f) + jH_q^*(f) \tag{4.129}$$

Lưu ý rằng, $\tilde{H}(f)$ và $\tilde{H}(-f)$ không phải là cặp liên hợp phức, vì tổng quát $\tilde{h}(t)$ không phải là hàm thực của biến thời gian. Lấy liên hợp phức của (4.129) ta có:

$$\tilde{H}^*(-f) = H_d(f) - jH_q(f) \tag{4.130}$$

Cộng (4.126) và (4.130) ta được:

$$H_d(f) = \frac{1}{2}\left(\tilde{H}(f) + \tilde{H}^*(-f)\right) \tag{4.131}$$

Nhân (4.130) với -1 và cộng kết quả với (4.126) ta có:

$$H_q(f) = \frac{1}{j2}\left(\tilde{H}(f) - \tilde{H}^*(-f)\right) \tag{4.132}$$

Tìm được hàm $h_d(t)$ và $h_q(t)$ bằng cách biến đổi Fourier ngược của $H_d(f)$ và $H_q(f)$. Nếu $\tilde{H}(f) = \tilde{H}^*(-f)$ sao cho $\tilde{H}_q(f)$ và $h_q(t)$ đều bằng 0, thì $\tilde{H}(f)$ được gọi là đối xứng liên hợp phức qua f_0. Hình 4.13 minh họa $\tilde{H}(f)$, $\tilde{H}^*(-f)$, $H_d(f)$ và $H_q(f)$ cho trường hợp $H(f)$ là bộ lọc thông dải lý tưởng.

Ví dụ 4.8: Ta xét việc xác định $h_d(t)$ và $h_q(t)$ trực tiếp từ $H(f)$. Từ $H(f)$, $\tilde{H}(f)$ được viết sử dụng (4.108) như được minh họa trong hình 4.13. Biến đổi Fourier ngược của $\tilde{H}(f)$ là:

$$\tilde{h}(t) = \int_{-\infty}^{\infty} \tilde{H}(f) e^{j2\pi ft}\, df = \int_{f_l - f_0}^{f_u - f_0} e^{j2\pi ft}\, df \tag{4.133}$$

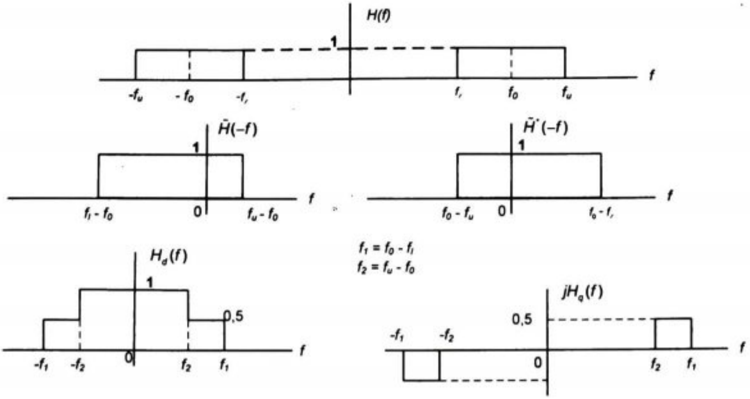
Vì $\tilde{H}(f) = 1$ trong khoảng lấy tích phân, nên ta có kết quả như sau:

$$\tilde{h}(t) = \frac{1}{j2\pi t}\left[e^{j2\pi(f_u - f_0)t} - e^{j2\pi(f_l - f_0)t}\right] \tag{4.134}$$

Phương trình trên có thể viết như sau

$$\tilde{h}(t) = \frac{1}{j2\pi t}\left[e^{j\pi(f_u - f_l)t} - e^{j\pi(f_l - f_u)t}\right].e^{-j2\pi\left(f_0 - \frac{f_u + f_l}{2}\right)t} \tag{4.135}$$

$$\tilde{h}(t) = \frac{1}{\pi t}\sin\left(\pi\left(f_u - f_l\right)t\right).e^{-j2\pi\left(f_0 - \frac{f_u + f_l}{2}\right)t} \tag{4.136}$$



Hình 4.13: $\tilde{H}(f)$, $\tilde{H}^*(-f)$, $H_d(f)$, $H_q(f)$ *đối với H(f) được giả định*

Lấy các phần thực và phần ảo ta có:

$$h_d(t) = \frac{1}{\pi t}\sin\left(\pi(f_u - f_l)t\right).\cos\left(2\pi\left(f_0 - \frac{f_u + f_l}{2}\right)t\right) \tag{4.137}$$

$$h_q(t) = -\frac{1}{\pi t}\sin\left(\pi(f_u - f_l)t\right).\sin\left(2\pi\left(f_0 - \frac{f_u + f_l}{2}\right)t\right) \tag{4.138}$$

Lưu ý rằng, nếu f_0 được chọn sao cho là tần số trung tâm $(f_u + f_l)/2$, thì $h_q(t) = 0, \forall t$. Điều này hiển nhiên là sẽ đơn giản hóa mô hình mô phỏng thông thấp được minh hoạ trong hình 4.12. Kết luận quan trọng là khối lượng tính toán để tìm tín hiệu ra của hệ thống khi biết tín hiệu vào được giảm đi 2 lần.

Như đã đề cập trong ví dụ trước, trong nhiều trường hợp thực tế f_0 có thể được chọn sao cho thỏa mãn $\tilde{h}_q(t) << \tilde{h}_d(t)$ với mọi giá trị của t. Trong các trường hợp như vậy thì $h_q(t)$ thường được bỏ qua, vì vậy đường bao phức của đáp ứng xung kim có thể được xấp xỉ như một hàm thực mà không mất đi tính chính xác. Như đã chỉ ra trong ví dụ trước, điều quan trọng là sử dụng ưu điểm của phép xấp xỉ khi ứng dụng, vì khi ta loại được $h_q(t)$ thì cũng giảm khối lượng tính toán của quá trình lọc 2 lần. Từ lý thuyết biến đổi Fourier cơ bản cho thấy $\tilde{h}(t)$ là thực nếu $\tilde{H}(f)$ có tính đối xứng liên hợp (phổ biên độ là hàm chẵn, phổ pha là hàm lẻ) qua $f = 0$. Đây cũng là trường hợp mà nếu hàm truyền đạt của bộ lọc thông dải $H(f)$ thể hiện tính chất đối xứng liên hợp qua $f = f_0$. Hầu hết các thiết kế bộ lọc thường *xấp xỉ tính chất* này *nếu* băng thông của bộ lọc nhỏ hơn so với tần số trung tâm của bộ lọc. Thành phần vuông pha có thể được xem là phép đo tính bất đối xứng liên hợp của *H(f)* qua f_0.

4.4. Tín hiệu đa sóng mang

Xét ghép kênh phân chia theo tần số (FDM) của M tín hiệu:

$$y(t) = \sum_{i=1}^{M} a_i(t)\cos\left[2\pi f_i t + \phi_i(t)\right] \tag{4.139}$$

Trong đó $a_i(t)$ và $\phi_i(t)$ biểu diễn cho điều chế pha và biên độ của sóng mang thứ i, và f_i là tần số sóng mang thứ i. Vì thành phần $a_i(t)$ là thực nên ta có thể viết:

$$y(t) = \mathrm{Re}\left\{\sum_{i=1}^{M} \underbrace{a_i(t)\mathrm{e}^{j\phi_i(t)}}_{\tilde{x}_i(t)}.\mathrm{e}^{j2\pi f_i t}\right\} \tag{4.140}$$

Định nghĩa:

$$\tilde{x}_i(t) = a_i(t).\mathrm{e}^{j\phi_i(t)} \tag{4.141}$$

Ta có:

$$y(t) = \mathrm{Re}\left\{\sum_{i=1}^{M} \tilde{x}_i(t)\mathrm{e}^{j2\pi f_i t}\right\} \tag{4.142}$$

Ta có thể định nghĩa đường bao phức của $y(t)$ là:

$$y(t) = \sum_{i=1}^{M} \tilde{y}_i(t)\mathrm{e}^{j2\pi f_0 t} \tag{4.143}$$

Trong đó, f_0 là lựa chọn tùy ý. Với định nghĩa này thì $y(t)$ có thể được viết như sau:

$$y(t) = \mathrm{Re}\left\{\sum_{i=1}^{M} \tilde{x}_i(t).\mathrm{e}^{j2\pi(f_i - f_0)t}.\mathrm{e}^{j2\pi f_0 t}\right\} \tag{4.144}$$

Vì vậy, đường bao phức của $y(t)$ là:

$$\begin{aligned}\tilde{y}(t) &= \sum_{i=1}^{M} \tilde{x}_i(t).\mathrm{e}^{j2\pi(f_i - f_0)t} \\ &= \sum_{i=1}^{M} a_i(t).\mathrm{e}^{j\phi_i(t)}.\mathrm{e}^{j2\pi(f_i - f_0)t}\end{aligned} \tag{4.145}$$

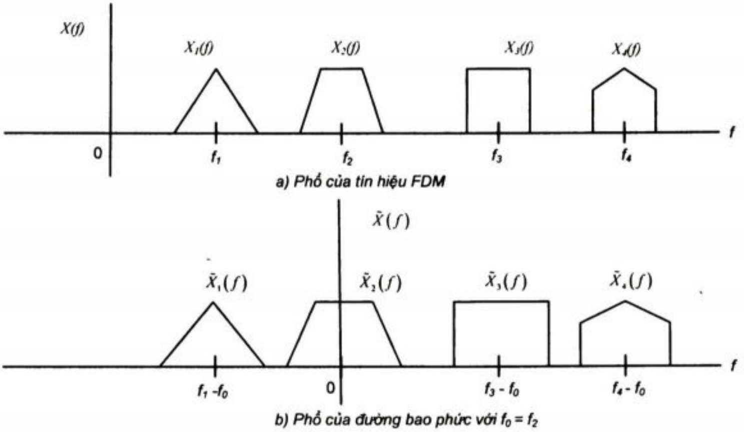
Vì vậy, các thành phần đồng pha và vuông pha của tín hiệu FDM được cho bởi:

$$y_d(t) = \sum_{i=1}^{M} a_i(t)\cos[2\pi(f_i - f_0)t + \phi_i(t)] \tag{4.146}$$

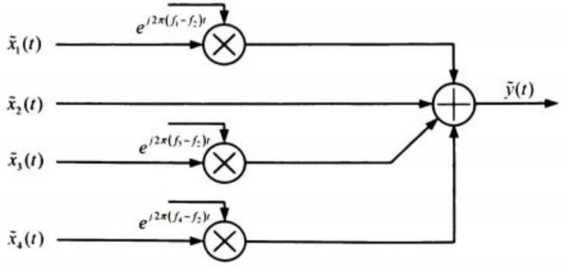
$$y_q(t) = \sum_{i=1}^{M} a_i(t)\sin[2\pi(f_i - f_0)t + \phi_i(t)] \tag{4.147}$$

Ví dụ 4.9: Xét tín hiệu ghép phân chia theo tần số gồm 4 kênh tín hiệu được cho ở hình 4.14(a). Giả sử tín hiệu quan tâm là $x_2(t)$ và sẽ thực hiện mô phỏng để kiểm tra các ảnh hưởng của *nhiễu* kênh lân cận và *méo* do điều chế ký sinh do tính phi tuyến của bộ khuếch đại gây ra. Vì quan tâm đến tín hiệu $x_2(t)$, nên phải đặt $f_0 = f_2$, sao cho $x_2(t)$ được dịch đến $f = 0$ khi đường bao phức của tín hiệu tổng hợp $y(t)$ được tạo ra. Với $f_0 = f_2$, thì đường bao phức của $y(t)$ là:

$$\tilde{y}(t) = \sum_{i=1}^{M} \tilde{x}_i(t).e^{j2\pi(f_i - f_2)t} \tag{4.148}$$



Hình 4.14: Đường bao phức của tín hiệu FDM



Hình 4.15: Xử lý tín hiệu cho đường bao phức của tín hiệu FDM

Điều này được minh họa trong hình 4.14(b) với $f_0 = f_2$. Lưu ý rằng tần số lấy mẫu nhỏ nhất đối với $\tilde{y}(t)$ phụ thuộc vào tín hiệu thông dải được dịch tới $f = 0$. Trường hợp $f_0 = f_2$, thì tần số lấy mẫu phải thỏa mãn:

$$f_s > 2\left[\left(f_4 - f_2\right) + \frac{B}{2}\right] \tag{4.149}$$

Trong đó B là băng thông của $X_4(f)$. Quá trình tạo ra $\tilde{y}(t)$ được minh họa trong hình 4.15.

4.5. Hệ thống phi tuyến và thay đổi theo thời gian

Chương này chủ yếu tập trung vào hệ thống cố định (bất biến) và tuyến tính. Nhiều hệ thống thực tế lại chứa các thành phần thay đổi theo thời gian (như kênh vô tuyến), các phần tử phi tuyến (như bộ khuếch đại công suất lớn hoạt động ở gần điểm bão hòa). Việc thiết kế và phân tích các hệ thống chứa các phần tử phi tuyến, phần tử biến đổi theo thời gian hay cả hai bằng các công cụ toán học truyền thống là rất khó thậm chí là không thể. Do đó, mô phỏng sẽ là một công cụ thường được sử dụng để thiết kế và phân tích các hệ thống này. Chương 12 tập trung xét hệ thống phi tuyến và chương 13 dành cho hệ thống thay đổi theo thời gian. Tuy nhiên, chương này cũng đề cập một cách vắn tắt hệ thống phi tuyến và phụ thuộc thời gian.

4.5.1. Hệ thống phi tuyến

Khái niệm cơ bản về hàm truyền đạt không được định nghĩa đối với hệ thống *phi tuyến*. Ở dạng tổng quát, quan hệ vào/ra của hệ thống phi tuyến không dùng được phép tích chập dù là đáp ứng xung kim có thể dùng cho hệ thống phi tuyến. Phép tích chập quen thuộc dựa trên khái niệm xếp chồng không dùng được đối với hệ thống phi tuyến. Tất nhiên, các mô hình mô phỏng cho các hệ thống phi tuyến có thể được triển khai nhưng thường được dựa trên đo đạc từ các hệ thống vật lý. Đôi khi vẫn dùng phân tích để triển khai các mô hình mô phỏng cho các hệ thống phi tuyến nhưng các kỹ thuật này thường là đặc biệt và không thể tổng quát hoá được. Một số mô hình mô phỏng quan trọng cho các hệ thống phi tuyến được triển khai ở chương 12.

Ta nghiên cứu mô hình mô phỏng cho một hệ thống phi tuyến đơn giản. Mô hình được triển khai trong ví dụ 4.10 là rất hữu hiệu cho việc nghiên cứu tiếp theo.

Ví dụ 4.10: Giả sử tín hiệu vào hệ thống có dạng:

$$x(t) = A(t)\cos[2\pi f_0 t + \theta(t)] \tag{4.150}$$

Các phép đo được thực hiện ở đầu ra của hệ thống với nhiều giá trị của $A(t)$ và $\phi(t)$ cho thấy rằng đường bao tín hiệu ra của hệ là không đổi và độc lập với $A(t)$ nhưng các điểm chéo không của tín hiệu vào được duy trì và phù hợp với điểm chéo không đầu ra hệ thống. Vì vậy, các phép đo cho thấy hệ thống có thể được mô hình hoá một cách chính xác bởi một bộ giới hạn cứng thông dải. Nhiệm vụ của ta là triển khai mô hình mô phỏng cho thiết bị đó.

Đường bao phức của tín hiệu vào hệ thống là:

$$\tilde{x}(t) = A(t)e^{j\theta(t)} = x_d(t) + jx_q(t) \tag{4.151}$$

Đầu ra của bộ giới hạn cứng thông dải được định nghĩa là tín hiệu hình sin có biên độ không đổi và độ lệch pha bằng độ lệch pha tín hiệu đầu vào. Vì vậy:

$$y(t) = B\cos[2\pi f_0 t + \theta(t)] \tag{4.152}$$

Đường bao phức tương ứng là:

$$\tilde{y}(t) = B\,e^{j\theta(t)} \tag{4.153}$$

Trong đó B là hằng số dương giả định. Ta thấy rằng, đối với bộ giới hạn cứng thông dải, nó là đường bao của tín hiệu bị giới hạn cứng chứ không phải là chính tín hiệu đó.

Lưu ý rằng theo định nghĩa về độ lớn:

$$\sqrt{x_d^2(t) + x_q^2(t)} = |A(t)| \tag{4.154}$$

Đường bao phức của tín hiệu ra được viết:

$$\tilde{y}(t) = \frac{B}{|A(t)|}\tilde{x}(t) \tag{4.155}$$

Vì vậy:

$$\tilde{y}(t) = \frac{B\left[x_d(t) + jx_q(t)\right]}{\sqrt{x_d^2(t) + x_q^2(t)}} \tag{4.156}$$

từ đó, ta có thành phần đồng pha và vuông pha là:

$$y_d(t) = \frac{Bx_d(t)}{\sqrt{x_d^2(t) + x_q^2(t)}} \tag{4.157}$$

$$y_q(t) = \frac{Bx_q(t)}{\sqrt{x_d^2(t) + x_q^2(t)}} \tag{4.158}$$

Thiết bị được định nghĩa bởi (4.157) và (4.158) được xem là bộ giới hạn cứng thông dải. Nó loại bỏ tất cả các thay đổi trên đường bao trong khi đó vẫn duy trì các vị trí chéo không.

Nhiều bộ giải điều chế làm việc dựa vào tính phi tuyến. Ví dụ, giả sử tín hiệu thu r(t) ở đầu vào bộ giải điều chế được xác định bởi:

$$r(t) = A(t)\cos[2\pi f_C t + \phi(t)] \tag{4.159}$$

Cũng giả thiết rằng, nhiệm vụ của bộ giải điều chế là khử phần dương của đường bao như trường hợp AM. Thường dùng bộ tách sóng đường bao để khôi phục đường bao (giải điều chế AM) có đầu ra z(t) được xác định bởi:

$$z(t) = |\tilde{r}(t)| = \left|A(t)e^{j\phi(t)}\right| \tag{4.160}$$

Nó là:

$$z(t) = |A(t)| \tag{4.161}$$

Cũng hữu hiệu để khôi phục sóng mang là bộ giải điều chế luật bình phương được định nghĩa bởi:

$$z(t) = \left|\tilde{r}(t)\right|^2 = \left|A(t)\right|^2 \tag{4.162}$$

Đường bao và các bộ giải điều chế luật bình phương là các ví dụ cho các bộ giải điều chế phi tuyến không nhất quán và được dùng khi không yêu cầu phải khôi phục độ lệch pha sóng mang $\phi(t)$.

Tồn tại nhiều tình huống phải khôi phục độ lệch pha sóng mang chẳng hạn các bộ giải điều chế tín hiệu PM và FM tương tự và các bộ giải điều chế tín hiệu số PSK và QPSK. Khối chức năng cơ bản phải có để khôi phục pha là vòng khoá pha PLL. Vòng khoá pha PLL là một hệ thống phi tuyến và được xét chi tiết ở chương 6.

4.5.2. Hệ thống thay đổi theo thời gian

Đối nghịch với trường hợp phi tuyến, nếu hệ thống là tuyến tính phụ thuộc thời gian, thì có nhiều công cụ trong chương này để phân tích và mô hình hoá hệ thống. Điều này là đúng vì chỉ cần hệ thống là tuyến tính, thì vẫn dùng được tích chập để xây dựng quan hệ vào/ra cho hệ thống trong miền thời gian và quan hệ vào ra của hệ thống trong miền tần số sử dụng hàm truyền đạt. Đáp ứng xung kim hệ thống và hàm truyền đạt hệ thống được định nghĩa cho các hệ thống tuyến tính phụ thuộc thời gian. Tuy nhiên, để tính đến bản chất thay đổi theo thời gian của hệ thống, cả đáp ứng xung kim hệ thống và hàm truyền đạt hệ thống phải được biến đổi so với các định nghĩa bất biến của nó.

Ví dụ: Nếu hệ thống tuyến tính thay đổi theo thời gian, thì quan hệ tín hiệu vào x(t) và tín hiệu ra y(t) được xác định bởi tích chập các đường bao phức, nghĩa là:

$$\tilde{y}(t) = \int_{\infty}^{\infty} \tilde{h}(\tau,t).\tilde{x}(t-\tau)d\tau \tag{4.163}$$

Trong đó $\tilde{h}(\tau,t)$ là đáp ứng xung kim phụ thuộc thời gian của hệ thống. Đáp ứng xung kim $\tilde{h}(\tau,t)$ được định nghĩa là đáp ứng của hệ thống được đo ở thời điểm t đối với một xung kim được đặt vào ở thời điểm sớm hơn τ giây. Nói cách khác, một xung kim được đặt vào hệ thống ở thời điểm $t - \tau$ và đáp ứng đó đo được ở thời điểm t, sau một khoảng "*thời gian trôi*" là τ. Đối với hệ thống bất biến, thì đáp ứng xung kim chỉ là một hàm của hiệu số thời gian $t - \tau$. Xung kim đó được coi là được đặt ở thời điểm $t - \tau = 0$ và đáp ứng xung kim là $h(\tau)$ quen thuộc.

Vì đáp ứng xung kim của hệ thống phụ thuộc thời gian là một hàm của hai biến miền thời gian t và τ, nên hàm truyền đạt của hệ thống phụ thuộc thời gian cũng là một hàm của hai biến miền tần số. Nó được định nghĩa là:

$$\tilde{H}(f_1,f_2) = \int_{\infty}^{\infty}\int_{\infty}^{\infty} \tilde{h}(\tau,t).e^{-j2\pi(f_1\tau+f_2t)}\,d\tau dt \tag{4.164}$$

Theo đó $\tilde{h}(\tau,t)$ & $\tilde{H}\left(f_1,f_2\right)$ là cặp chuyển đổi Fourier

Các khái niệm này sẽ được triển khai ở chương 13, các kết quả sẽ là nền tảng lý thuyết để thiết kế, phân tích và mô phỏng các hệ thống tuyến tính phụ thuộc thời gian. Vì vậy, mặc dù

chương này chủ yếu tập trung vào hệ thống tuyến tính bất biến nhưng nhiều khái niệm được đề cập sẽ là cơ sở để phát triển phương pháp luận cho việc mô phỏng các hệ thống thay đổi theo thời gian.

Ví dụ 4.11: Xét tình huống được minh họa ở hình 4.16. Một máy thu trong trạng thái chuyển động thu một tín hiệu từ một máy phát được truyền lan theo hai đường truyền. Một đường truyền lan trực tiếp từ máy phát đến máy thu di động. Đường thứ hai phản xạ từ toà nhà. Mô hình truyền lan này được gọi là mô hình truyền lan hai tia. Máy thu di động đang di chuyển. Hậu quả của việc di chuyển là gây ra độ dài truyền dẫn của cả hai đường truyền thay đổi theo thời gian. Hiển nhiên, cả suy hao tín hiệu lẫn trễ truyền lan đối với mỗi đường truyền đều thay đổi theo thời gian. Giả sử, trễ truyền sóng và suy hao đối với đường dẫn thứ n được ký hiệu là $a_n(t)$ và $\tau_n(t)$, thì thí hiệu thu được xác định bởi mô hình kênh là:

$$y(t) = a_1(t).x(t-\tau_1(t)) + a_2(t).x(t-\tau_2(t)) = \sum_{n=1}^{2} a_n(t).x\big(t-\tau_n(t)\big) \qquad (4.165)$$

Ta giả thiết rằng, đưa vào kênh là tín hiệu điều chế tổng quát:

$$x(t) = A(t).\cos\big(2\pi f_c t + \phi(t)\big) \qquad (4.166)$$

Thế (4.166) vào (4.165) nhận được:

$$y(t) = \sum_{n=1}^{2} a_n(t).A\big(t-\tau_n(t)\big).\cos\Big[2\pi f_c\big(t-\tau_n(t)\big) + \phi\big(t-\tau_n(t)\big)\Big] \qquad (4.167)$$

Vì mô phỏng mức dạng sóng thường được thực hiện bằng cách dùng các tín hiệu đường bao phức, nên ta phải xác định đường bao phức cho cả tín hiệu *x(t)* và *y(t)*.

Đường bao phức của tín hiệu phát là:

$$\tilde{x}(t) = A(t)e^{\phi(t)} \qquad (4.168)$$

Ta xác định đường bao phức của tín hiệu thu được định nghĩa bởi (4.167). Vì $a_n(t)$ và A(t) đều là giá trị thực nên (4.167) được viết lại là:

$$y(t) = \text{Re}\left\{\sum_{n=1}^{2} a_n(t).A\big(t-\tau_n(t)\big).e^{j\phi(t-\tau_n(t))}.e^{-j2\pi f_c \tau_n(t)}.e^{j2\pi f_c t}\right\} \qquad (4.169)$$

Từ (4.168):

$$\tilde{x}\big(t-\tau_n(t)\big) = A\big(t-\tau_n(t)\big).e^{j\phi(t-\tau_n(t))} \qquad (4.170)$$

Vì vậy:

$$y(t) = \text{Re}\left\{\sum_{n=1}^{2} a_n(t)\tilde{x}\big(t-\tau_n(t)\big).e^{-j2\pi f_c \tau_n(t)}.e^{j2\pi f_c t}\right\} \qquad (4.171)$$

Suy hao đường dẫn phức được định nghĩa là:

$$\tilde{a}_n(t) = a_n(t).e^{-j2\pi f_c \tau_n(t)} \qquad (4.172)$$

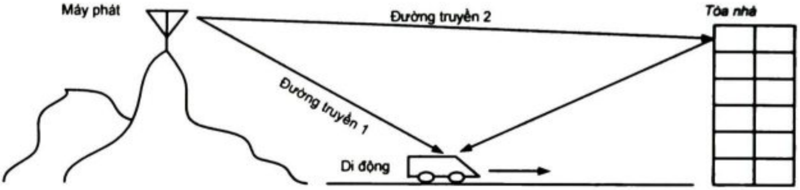
Dẫn đến:

$$y(t) = \text{Re}\left\{\sum_{n=1}^{2}\tilde{a}_n(t).\tilde{x}\left(t-\tau_n(t)\right).e^{-j2\pi f_c t}\right\} \tag{4.173}$$

Vì vậy, đường bao phức tín hiệu vào máy thu là:

$$\tilde{y}(t) = \sum_{n=1}^{2}\tilde{a}_n(t).\tilde{x}\left(t-\tau_n(t)\right) \tag{4.174}$$

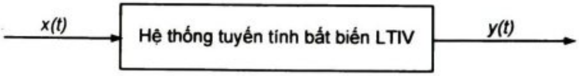
Điều này xác định mô hình kênh thông thấp phức. Mô hình này sẽ lại được xét đến ở chương 14 ở đó xét các mô hình kênh dạng sóng chi tiết hơn.



Hình 4.16: Minh họa kịch bản truyền thông di động hai tia

4.6. Minh họa tín hiệu và hệ thống tuyến tính bất biến

Nếu cho tín hiệu tuần hoàn *x(t)* qua hệ LTIV như hình 4.17, thì tín hiệu ra *y(t)* cũng là tín hiệu chu kỳ, có cùng tần số với tín hiệu vào, vì vậy có khai triển chuỗi Fouriers.



Hình 4.17: Tín hiệu chu kỳ cho qua hệ thống tuyến tính bất biến LTIV

Nếu *x(t)* và *y(t)* được khai triển là:

$$x(t) = \sum_{n=-\infty}^{\infty} x_n.e^{j2\pi\frac{n}{T_0}t} \tag{4.175}$$

$$y(t) = \sum_{n=-\infty}^{\infty} y_n.e^{j2\pi\frac{n}{T_0}t} \tag{4.176}$$

Thì có thể tìm được quan hệ giữa các hệ số chuỗi Fouriers của *x(t)* và *y(t)* bằng cách dùng tích chập.

$$\begin{aligned} y(t) &= \int_{-\infty}^{\infty} x(t-\tau)h(\tau)d\tau = \int_{-\infty}^{\infty}\sum_{n=-\infty}^{\infty} x_n.e^{j2\pi\frac{n}{T_0}(t-\tau)}h(\tau)d\tau \\ &= \sum_{n=-\infty}^{\infty} x_n\left(\int_{-\infty}^{\infty} h(\tau)e^{-j2\pi\frac{n}{T_0}\tau}d\tau\right)e^{j2\pi\frac{n}{T_0}t} \\ &= \sum_{n=-\infty}^{\infty} y_n e^{j2\pi\frac{n}{T}t} \end{aligned} \tag{4.178}$$

Từ mối quan hệ trước ta có (xem Phụ lục 4B) có trên đĩa CD.

$$y_n = x_n H\left(\frac{n}{T_0}\right) \tag{4.179}$$

Trong đó *H(f)* là hàm truyền đạt của hệ LTIV được cho bởi biến đổi Fourier của đáp ứng xung của nó *h(t)*:

$$H(f) = \int_{-\infty}^{\infty} h(t).e^{-j2\pi ft}dt \tag{4.180}$$

Ví dụ 4.12: Lọc các tín hiệu chu kỳ

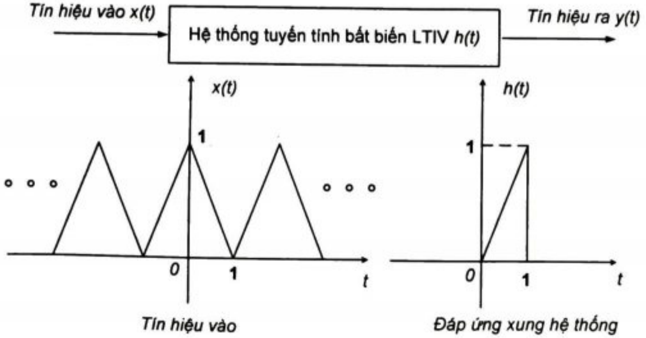
Cho chuỗi xung tam giác x(t) có chu kỳ $T_0 = 2$ được xác định trong một chu kỳ như

$$\Lambda(t) = \begin{cases} t+1, & -1 \le t \le 0 \\ -t+1, & 0 \le t \le 1 \\ 0 & \text{nếu khác} \end{cases}$$

Hãy, (1) xác định hệ số Fourier của *x(t)*; (2) vẽ phổ rời rạc của *x(t)*; (3) giả sử cho tín hiệu này qua hệ thống LTIV có đáp ứng xung được cho bởi:

$$h(t) = \begin{cases} t, & 0 \le t < 1 \\ 0, & \text{nếu khác} \end{cases}$$

Vẽ phổ rời rạc và tín hiệu đầu ra y(t). Biết rằng, dạng sóng của *x(t)* và *h(t)* được cho bởi hình 4.18.



Hình 4.18: Tín hiệu vào x(t) và đáp ứng xung của hệ thống h(t)

Hệ số của *x(t)* được xác định bởi:

$$x_n = \frac{1}{T_0}\int_{-T_0/2}^{T_0/2} x(t)e^{-j2\pi nt/T_0}dt = \frac{1}{2}\int_{-1}^{1}\Lambda(t)e^{-j\pi nt}dt = \frac{1}{2}\int_{-\infty}^{\infty}\Lambda(t)e^{-j\pi nt}dt$$

$$= \frac{1}{2}\Im\left[\Lambda(t)\right]_{f=n/2} \qquad = \frac{1}{2}SinC^2\left(\frac{n}{2}\right)$$

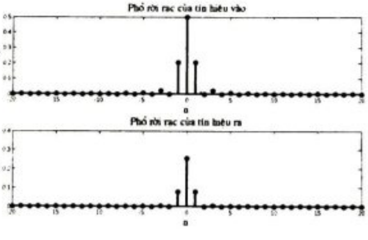
Trong đó ta đã khử khoảng bên ngoài [-1,1], biến đổi Fourier của *Λ(t)* là $SinC^2(f)$. **Kết** quả này cũng có thể đạt được bằng cách dùng biểu thức *Λ(t)* và thực hiện lấy tích phân từng phần theo định nghĩa FT thông thường. Thấy rõ, $x_n = 0$ *với* ∀ giá trị *n* chẵn trừ $n = 0$.

Phổ rời rạc của *x(t)* và phổ rời rạc của *y(t)* được kết quả chạy chương trình (hình 4.19).

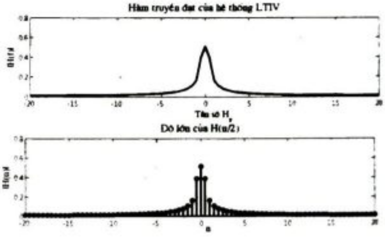
Trước hết, phải xuất phát từ hàm truyền đạt của hệ thống H(f). Mặc dù có thể tìm được bằng phương pháp giải tích, nhưng sẽ làm theo phương pháp số, kết quả phổ biên độ của hàm truyền đạt là $H\left(n/T_0\right) = H\left(n/2\right)$ được cho ở hình 4.20. Sau đó, tìm phổ rời rạc đầu ra y(t) bằng quan hệ :

$$y_n = x_n H\left(\frac{n}{T_0}\right) = \frac{1}{2}SinC^2\left(\frac{n}{2}\right)H\left(\frac{n}{2}\right)$$

Kết quả phổ rời rạc đầu ra được cho ở hình 4.19 kết quả chạy chương trình. Chương trình Matlab được cho ở file **NVD4_CS14.m**.



Hình 4.19: Phổ rời rạc của tín hiệu vào và ra



Hình 4.20: Hàm truyền đạt của hệ thống LTI V và độ lớn của H(n/2)

Một số bài tập cơ bản về: *(i)* chuỗi Fourier; *(ii)* biến đổi Fourier; *(iii)* công suất và năng lượng tín hiệu; *(iv)* khảo sát tín hiệu và hệ thống tuyến tính bất biến được cho ở dạng ví dụ có trong Phụ lục 4B.

Chương 5

MÔ HÌNH BỘ LỌC VÀ KỸ THUẬT MÔ PHỎNG

5.1. Mở đầu

Chương 5 tập trung triển khai các mô hình mô phỏng cho bộ lọc. Bộ lọc là một bộ phận quan trọng của rất nhiều phân hệ trong hệ thống tổng thể. Để mô phỏng, ta phải chuyển bộ lọc tương tự thành bộ lọc số một cách phù hợp. Có nhiều kỹ thuật khả dụng, chúng đều thực hiện xấp xỉ và đều gây ra lỗi trong kết quả mô phỏng. Chương này sẽ khai thác các kỹ thuật hữu hiệu nhất để tổng hợp và mô phỏng bộ lọc. Ngoài ra, cũng đề cập các hạn chế và nguồn lỗi.

Theo định nghĩa và phân loại, thì bộ lọc có tính chọn lọc tần số và đáp ứng xung kim thuộc loại đáp ứng xung kim hữu hạn FIR hoặc đáp ứng xung kim vô hạn IIR. Do tính chọn lọc tần số nên nó có tính nhớ, tín hiệu ra của bộ lọc tại thời điểm hiện tại được tính từ tín hiệu vào hiện thời và trước đó. Theo đó, bộ lọc cần có bộ lưu trữ, việc lưu trữ và lấy các giá trị mẫu làm tăng đáng kể mức độ tính toán trong chương trình mô phỏng, đồng nghĩa với tăng thời gian mô phỏng. Vì vậy, cần có các thuật toán sao cho giảm tải tính toán.

Chương này không trình bày chi tiết kỹ thuật thiết kế bộ lọc số (nhiều giáo trình đã đề cập thiết kế bộ lọc số và các kỹ thuật cơ bản đã được sử dụng trong nhiều năm) mà chỉ tóm các kỹ thuật hữu hiệu nhất và nhấn mạnh *lỗi lấy xấp xỉ* trong mô phỏng.

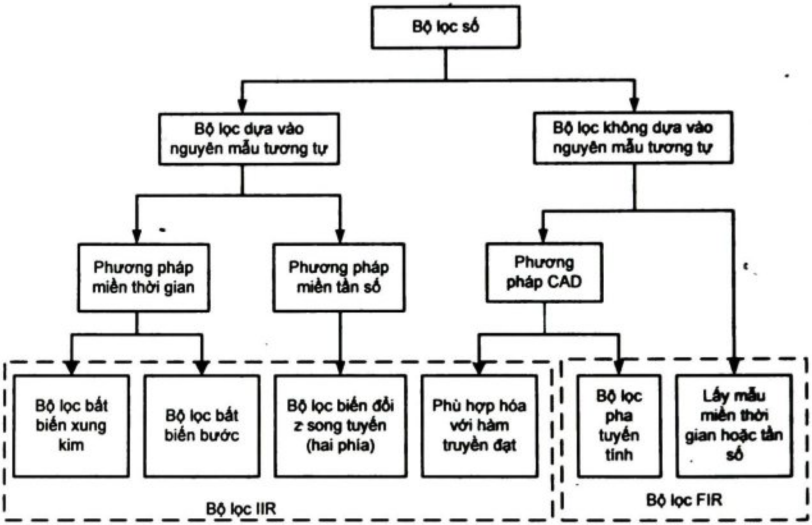
Tổng hợp một số kỹ thuật triển khai các bộ lọc số được minh họa ở hình 5.1. Các kỹ thuật truyền thống thường dựa trên các nguyên mẫu tương tự. Khi này, triển khai bộ lọc số trong mô hình mô phỏng bắt đầu bằng hàm truyền đạt miền s (Laplace) của bộ lọc tương tự từ đó tìm phần tử số tương đương. Sau đó, bài toán được quy về tìm bộ lọc số tương đương phù hợp với nguyên mẫu bộ lọc tương tự. Nhiều phương pháp nền tảng dựa trên đánh giá tính tương đương trong tiêu chuẩn miền thời gian hoặc tần số. Mặc dù, thường lọc theo các *đặc tính chọn lọc tần số* nhưng khi dùng tiêu chuẩn miền thời gian dẫn đến hiệu quả. Tiêu chuẩn miền thời gian được áp dụng bằng cách qui định đầu ra bộ lọc số phù hợp với đầu ra được lấy mẫu của nguyên mẫu tương tự tương ứng. Hai kỹ thuật tổng hợp nền tảng dựa trên tiêu chuẩn miền thời gian là *bộ lọc số bất biến xung kim* và *bộ lọc số bất biến bước*, trong đó: *(i) Bộ lọc số bất biến xung kim* là một thiết kế trong đó đáp ứng xung kim của bộ lọc số phù hợp với đáp ứng xung kim của nguyên mẫu tương tự được lấy mẫu; *(ii) Bộ lọc số bất biến bước*, đáp ứng *bước* của bộ lọc số phù hợp với *đáp ứng bước* của nguyên mẫu tương tự được lấy mẫu.

Nhiều thiết kế khác sử dụng "*các tín hiệu kiểm tra*" chứ không phải là các bước đơn vị hay các bước xung kim. Ta sẽ thấy, nếu nguyên mẫu tương tự và bộ lọc số là tương đương nhau trong miền thời gian thì cũng tương đương trong miền tần số, ít nhất đối với các giá trị tần số nhỏ so với tần số lấy mẫu.

Có lẽ phương pháp phổ biến nhất để ánh xạ một nguyên mẫu tương tự thành số là thông qua biến đổi z song tuyến tính (hai phía). Ở đây sử dụng biến đổi z song tuyến có hàm ý hướng

vào ứng dụng, hàm ý về cấu trúc và nguyên lý hoạt động hơn là mô tả toán học thuần túy. Phương pháp tổng hợp biến đổi z-hai phía (kỹ thuật đại số hoàn toàn) được thực thi sao cho cả nguyên mẫu tương tự và bộ lọc số có cùng đáp ứng tần số (pha và độ lớn) tại các giá trị tần số thiết kế cụ thể. Ngoài ra, phương pháp biến đổi z-hai phía khử các lỗi chồng phổ ở vùng mở rộng phổ tần do tính phi tuyến.

Kỹ thuật tổng hợp dựa trên nguyên mẫu tương tự tạo ra bộ lọc số IIR (đáp ứng xung kim vô hạn). Đáp ứng xung kim được tạo ra bởi một trong các phương pháp thiết kế IIR tiêu chuẩn được cắt ngắn để tạo thành bộ lọc FIR. Có thể giảm lỗi do việc cắt ngắn xuống mức chấp nhận bằng cách tính đến số lượng lớn thành phần trong đáp ứng xung kim của bộ lọc FIR kết quả.



Hình 5.1: Phân loại bộ lọc số

Thuộc tính quan trọng của bộ lọc số là có thể triển khai bộ lọc mà không cần có nguyên mẫu tương tự. Hầu hết các bộ lọc số quan trọng thuộc loại này, là các bộ lọc cho phép xấp xỉ hóa với một đáp ứng biên độ cho trước mà vẫn duy trì được đáp ứng pha pha tuyến tính một cách hoàn hảo. Đây là bộ lọc FIR được thực hiện theo cấu trúc đường trễ rẽ nhánh TDL. Tồn tại nhiều kỹ thuật thiết kế cho các bộ lọc này. Phương pháp nền tảng nhất là khai triển đáp ứng biên độ, là tuần hoàn theo tần số lấy mẫu trong chuỗi Fourier. Các hệ số Fourier xác định đáp ứng xung kim của bộ lọc số. Có thể dùng FFT để thực hiện quá trình này. Đây là một ví dụ về *lấy mẫu tần số*, bởi lẽ đáp ứng tần số được *"được lấy mẫu"* tại nhiều điểm khác nhau trong miền tần số. Thực hiện IFFT lên các mẫu tần này cho ta đáp ứng xung kim của bộ lọc. Lấy tích chập tín hiệu vào bộ lọc với đáp ứng xung kim (thực hiện mô hình mô phỏng của bộ lọc) cho ta tín hiệu đầu ra của bộ lọc.

Tồn tại nhiều kỹ thuật thiết kế được hỗ trợ bởi máy tính CAD để thiết kế các bộ lọc số. Trong chương này ta khai thác hai trong số chúng. Kỹ thuật đầu tiên dẫn đến bộ lọc IIR, kỹ thuật thứ hai dẫn đến bộ lọc FIR pha tuyến tính.

Có thể tham khảo phần Phụ lục 5B "Practical FIR Filter Design in MATLAB" để được rõ hơn về việc thiết kế bộ lọc FIR thực tế.

5.2. Bộ lọc IIR và FIR

Như đã đề cập, ta thường phân loại bộ lọc số theo khoảng thời gian đáp ứng xung kim (FIR hay IIR). Gắn liền với phân loại đáp ứng xung kim là thực hiện cấu trúc. Tại đây ta xét các mô hình bộ lọc khác nhau.

5.2.1. Bộ lọc IIR

Một bộ vi xử lý tín hiệu số tuyến tính (bộ lọc số) tính tín hiệu ra hiện thời $y[n]$ là tổng trọng số của N mẫu đầu ra trước $y[n-k]$, $1 \le k \le N$, mẫu đầu vào hiện thời $x[n]$ và N mẫu vào trước $x[n-k]$, $1 \le k \le N$. Nói cách khác, thuật toán để tính đầu ra hiện thời theo các đầu vào/ra trước đó là:

$$y[n] = \sum_{k=0}^{N} b_k x[n-k] - \sum_{k=1}^{N} a_k y[n-k] \tag{5.1}$$

Tồn tại nhiều thuật toán hiệu quả để thực hiện (5.1) trong chương trình mô phỏng.

Thấy rõ, nếu một trọng số nào đó trong (5.1), b_k hay a_k, $k \ge 1$ khác không thì bộ vi xử lý có tính nhớ, có tính chọn lọc tần số và được coi là một bộ lọc. *Trường hợp hệ thống thay đổi theo thời gian, thì một hoặc một vài trọng số sẽ là hàm của chỉ số n.* Hàm truyền đạt *H(z)* là biến đổi z hai phía của (5.1). Cần lưu ý rằng: *(i)* biến đổi z là phép toán tuyến tính vì vậy biến đổi của một tổng là tổng của các biến đổi; *(ii)* khi làm trễ *k* mẫu là tương đương với việc nhân với z^{-k}. Dẫn đến:

$$Y[z]\left[1 + \sum_{k=1}^{N} a_k z^{-k}\right] = X(z)\sum_{k=0}^{N} b_k z^{-k} \tag{5.2}$$

Theo đó, ta có hàm truyền đạt:

$$H[z] = \frac{Y[z]}{X(z)} = \frac{\sum_{k=0}^{N} b_k z^{-k}}{\left[1 + \sum_{k=1}^{N} a_k z^{-k}\right]} \tag{5.3}$$

Đây là dạng tổng quát của bộ lọc *tuyến tính bất biến.*

Trong các ứng dụng được xét ở đây, ta quan tâm đáp ứng xung kim và đáp ứng tần số của của bộ lọc. Đáp ứng xung kim $h[n]$ là biến đổi z ngược của hàm truyền đạt *H(z)*. Tìm đáp ứng tần số bằng cách thay $z = e^{j2\pi fT}$ vào hàm truyền đạt. Nói cách khác, đáp ứng tần số là:

$$H(z)\big|_{z=e^{j2\pi fT}} = H\left(e^{j2\pi fT}\right) \tag{5.4}$$

Tìm được đáp ứng xung kim của bộ lọc số $h[n]$ bằng cách đặt $x[n] = \delta[n]$, với:

$$\delta[n]=\begin{cases}1, & n=0\\ 0, & n \neq 0\end{cases} \tag{5.5}$$

vào (5.1). Do bản chất đệ qui của (5.1) (nghĩa là $y[n]$ là *hàm* của $y[n-1]$) nên ở dạng tổng quát đáp ứng xung kim $h[n]$ sẽ có khoảng thời gian vô hạn và được gọi bộ lọc đáp ứng xung kim vô hạn IIR. Cần lưu ý rằng, đáp ứng xung kim $h[n]$ là hàm rời rạc của chỉ số n, trong khi đáp ứng tần số là một hàm liên tục của biến tần số liên tục *f*.

5.2.2. Bộ lọc FIR

Có được bộ lọc FIR nếu $a_k = 0, k \geq 1$ trong (5.1) (tương đương trong (5.3)) Theo đó, hàm truyền đạt và đáp ứng xung kim của bộ lọc FIR là:

$$H[z]=\sum_{k=0}^{N} b_k z^{-k} \tag{5.6}$$

$$h[n]=\sum_{k=0}^{N} b_k.\delta[n-k] \tag{5.7}$$

khác không chỉ khi $0 \leq n \leq N$. Do đó, đáp ứng xung kim có nhiều nhất $N+1$ thành phần khác không và có khoảng thời gian hữu hạn. Thuật toán để tạo chuỗi đầu ra bộ lọc $y[n]$ từ chuỗi đầu vào $x[n]$ là một tổng chập:

$$y[n]=\sum_{k=0}^{N} b_k.x[n-k] = \sum_{k=0}^{N} h[k].x[n-k] \tag{5.8}$$

Suy ra trực tiếp từ (5.1) khi $a_k = 0, k > 0$

5.2.3. Tổng hợp và mô phỏng

Cần phải chỉ ra rằng, bộ lọc trong chương trình mô phỏng gồm hai phép toán rất khác biệt: *(i) Phép toán thứ nhất* là *tổng hợp*, ta phải xác định rõ các yêu cầu lọc, hàm truyền đạt của bộ lọc H(z), hàm H(z) đáp ứng các yêu cầu lọc đã định. Điều này thiết lập mô hình mô phỏng. Kết quả của quá trình tổng hợp thường được biểu diễn ở dạng 2 véc-tơ, một véc-tơ chứa các hệ số ở mẫu a_k, một véc-tơ chứa các hệ số ở tử số b_k. Hai véc-tơ này sẽ tạo hàm truyền đạt (5.3) và thuật toán tạo đầu ra bộ lọc khi biết đầu vào. Khối lượng tính toán của quá trình tổng hợp không quá lớn, thậm chí khi sử dụng thuật toán phức tạp, vì tổng hợp chỉ thực hiện duy nhất 1 lần do vậy nó nằm ngoài vòng lặp mô phỏng chính; *(ii) Phép toán thứ hai* là tính tín hiệu ra bộ lọc tại mỗi bước thời gian mô phỏng (nghĩa là tại mỗi nhịp của đồng hồ mô phỏng). Quá trình này nhiều khi lặp lại hàng triệu lần, thậm chí là hàng tỉ lần trong chương trình mô phỏng Monte Carlo. Do vậy, khối lượng tính toán cho quá trình này phải được giảm thiểu để thời gian chạy mô phỏng hợp lý. Cấu trúc bộ lọc chuyển vị trong mục 5.3 dành cho chủ đề này.

5.3. Thực hiện bộ lọc IIR và FIR

Ta khảo sát vắn tắt cách thực thi bộ lọc số trong chương trình mô phỏng. Như đã đề cập, với mục đích giảm thiểu tải tính toán để thời gian chạy chương trình mô phỏng là nhỏ nhất.

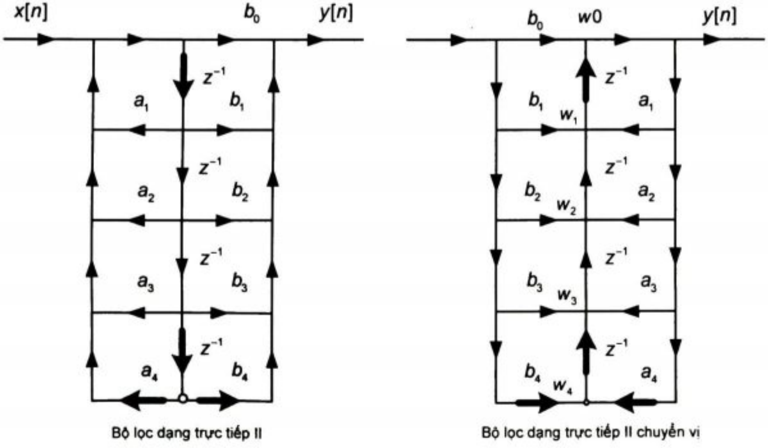
5.3.1. Thực hiện dạng trực tiếp II và dạng trực tiếp II chuyển vị

Kỹ thuật hiệu quả để thực thi bộ lọc số IIR trong mô phỏng là kiến trúc dạng trực tiếp chuyển vị II. Biểu đồ dòng tín hiệu của kiến trúc dạng trực tiếp II chuyển vị và kiến trúc dạng trực tiếp II được minh họa trong hình 5.2. Ta bắt đầu với kiến trúc dạng trực tiếp II vì nó dễ dàng thực thi phương trình sai phân (5.1). Ta nên dành thời gian để kiểm tra cả hai kiến trúc được minh họa trong hình 5.2, chúng đều thỏa mãn công thức (5.1) và (5.3).

Cấu trúc dạng trực tiếp II chuyển vị được dùng phổ biến nhất trong mô phỏng bộ lọc vì tốc độ thực hiện. Dễ dàng rút ra từ cấu trúc dạng trực tiếp II. Nguyên tắc tạo cấu trúc bộ lọc chuyển vị từ một cấu trúc bộ lọc cho trước như sau

1. Kéo lại các đồ hình luồng tín hiệu (dạng trực tiếp II) gốc duy trì kiến trúc (tất cả các liên kết vẫn duy trì các vị trí tương ứng của nó).
2. Đảo hướng luồng tín hiệu trên mỗi liên kết.
3. Gán cho liên kết mới cùng một toán hạng (nhân với hằng số, trễ,...), các toán hạng này đã được gán cho liên kết gốc.
4. Nếu muốn, lật (trái sang phải) đồ hình luồng tín hiệu mới sao cho hướng của luồng tín hiệu vào/ra phù hợp với đồ hình luồng tín hiệu gốc (Lưu ý: luồng tín hiệu thường từ trái qua phải).

Đồ hình luồng tín hiệu mới được gọi là cấu trúc dạng trực tiếp II chuyển vị (DF II) sẽ có cùng hàm truyền đạt với đồ hình luồng tín hiệu gốc.



Hình 5.2: Các cấu trúc thực thi các bộ lọc IIR

Để thấy được tính hấp dẫn của việc thực thi dạng trực tiếp II chuyển vị, ta xét bộ lọc dạng trực tiếp II (DF II) chuyển vị bậc 4 được cho ở hình 5.2 (mở rộng cho các bậc cao hơn cũng

tương tự). Ta đưa mẫu tín hiệu vào *x[n]* để tính tín hiệu ra *y[n]*. Trước tiên là tính biến trạng thái $w_j[n]$ với j = 0, 1,..., 4. Lưu ý rằng, với bộ lọc bậc 4 có 5 biến trạng thái trong công thức. Năm biến trạng thái này được cho bởi:

$$w_0[n] = w_1[n-1] + b_0 x[n] \tag{5.9}$$

$$w_1[n] = -a_1 w_0[n] + w_2[n-1] + b_1 x[n] \tag{5.10}$$

$$w_2[n] = -a_2 w_0[n] + w_3[n-1] + b_2 x[n] \tag{5.11}$$

$$w_3[n] = -a_3 w_0[n] + w_4[n-1] + b_3 x[n] \tag{5.12}$$

$$w_4[n] = -a_4 w_0[n] + b_4 x[n] \tag{5.13}$$

Xét ưu điểm tính toán của kiến trúc chuyển vị được định nghĩa bởi các phương trình trên.

Có thể tính toán các biến trạng thái như được biểu diễn bởi (5.9) đến (5.13) "trong chuỗi". Ví dụ, cho tín hiệu vào *x[n]* thì tính được $w_0[n]$, vì $w_1[n-1]$ đã được biết từ trước thông qua vòng lặp mô phỏng. Một khi $w_0[n]$ đã biết thì tính được $w_1[n]$. Tiếp tục tính toán, ta thấy rằng $w_j[n]$ chỉ phụ thuộc vào $w_k[n]$ trong đó $k<j$. Vì vậy, quá trình tính toán mỗi biến trạng thái chỉ yêu cầu biết các đại lượng tính toán trước. Đoạn mã chương trình Matlab thực hiện (5.9) đến (5.13) trong vòng lặp mô phỏng được cho dưới đây.

```
w1 = 0; w2 = 0; w3 = 0 w4 = 0; % Khëi t¹o c¸c biÕn tr¹ng th¸i
for k = 1:nputs
 ...
w0 = w1 + b0*x
 w1 = -a1*w0 + w2 + b1*x;
 w2 = -a2*w0 + w3 + b2*x;
 w3 = -a3*w0 + w4 + b3*x;
 w4 = -a4*w0 + b4*x;
 y = w0;
 ...
end
```

Trong đoạn mã trên *x* và *y* là tín hiệu vào/ra bộ lọc hiện thời. Lưu ý rằng, các toán hạng trễ một mẫu được minh họa trong (5.9) tới (5.13) được thực hiện theo thứ tự trong đó các biến trạng thái được tính toán. Không cần thiết phải lưu trữ và khôi phục. Kết quả: thuật toán nhanh hơn so với các thuật toán dựa trên các cấu trúc khác, đây là lý do tại sao bộ lọc thường trình (*routine filter*) của Matlab dựa trên cấu DF II chuyển vị. Cũng lưu ý, các biến trạng thái w_1, w_2, w_3, w_4 phải được khởi tạo trước khi đi vào trong vòng lặp mô phỏng lần đầu tiên. Việc khởi tạo này tạo ra một đáp ứng tạm thời đầu ra của bộ lọc. Cụ thể, vòng lặp mô phỏng phải được thực hiện nhiều lần trước khi số liệu hữu hiệu được tập hợp từ quá trình mô phỏng. Thời gian này được coi là "*thời gian định cư*" và gấp vài lần tỉ lệ nghịch băng thông bộ lọc. Dạng véc-tơ của các tính toán trên được đề cập ngắn ngọn (xem ví dụ 5.1).

Các phương trình trạng thái cho bộ lọc số thường được biểu diễn ở dạng ma trận. Dạng của ma trận biến trạng thái rất thuận lợi khi bộ lọc có nhiều đầu vào/ra. Tuy nhiên ở đây, ta chỉ quan tâm đến bộ lọc có một đầu vào *x[n]* và một đầu ra *y[n]*. Biểu thức chung cho bộ lọc 1 đầu vào là:

$$\mathbf{W}[n] = \mathbf{F_c W}[n] + \mathbf{F_d W}[n-1] + \mathbf{Bx}[n] \tag{5.14}$$

Trong đó $\mathbf{W}[n]$ và $\mathbf{W}[n-1]$ là các véc-tơ cột $k \times 1$ thể hiện biến trạng thái trước và hiện tại, **Fc** và **Fd** là các ma trận hệ số và **B** là véc-tơ cột $k \times 1$ ghép đầu vào *x[n]* với các biến trạng thái, phương trình đầu ra đối với một đầu ra là:

$$\mathbf{y}[n] = C\mathbf{W}[n] \tag{5.15}$$

Trong đó **C** là véc-tơ hàng $1 \times k$.

Các phương trình (5.9) tới (5.13) được viết ở dạng ma trận là:

$$\begin{bmatrix} w_0[n] \\ w_1[n] \\ w_2[n] \\ w_3[n] \\ w_4[n] \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -a_1 & 0 & 0 & 0 & 0 \\ -a_2 & 0 & 0 & 0 & 0 \\ -a_3 & 0 & 0 & 0 & 0 \\ -a_4 & 0 & 0 & 0 & 0 \end{bmatrix} . \begin{bmatrix} w_0[n] \\ w_1[n] \\ w_2[n] \\ w_3[n] \\ w_4[n] \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_0[n-1] \\ w_1[n-1] \\ w_2[n-1] \\ w_3[n-1] \\ w_4[n-1] \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} x[n] \tag{5.16}$$

Cho thấy rằng, có thể tính toán các trạng thái "*theo chuỗi*", vì ma trận $\mathbf{F}_c$ thiết lập quan hệ **W**j[n] và **W**k[n] có giá trị không ở trên và ở bên trên đường chéo chính. Biểu đồ dòng tín hiệu có tính chất này được gọi là biểu đồ khả tính toán. Thuật ngữ *khả tính toán* ở đây không có nghĩa là cần phải có dạng này để tính đầu ra khi cho trước cho một đầu vào. Thấy rõ, vì cấu trúc tam giác phía trên có thể được loại bỏ bằng cách đánh nhãn lại các trạng thái. Hơn nữa, thuật ngữ *khả tính toán* có nghĩa là các trạng thái có thể được tính toán lần lượt như được cho từ (5.9) đến (5.13).

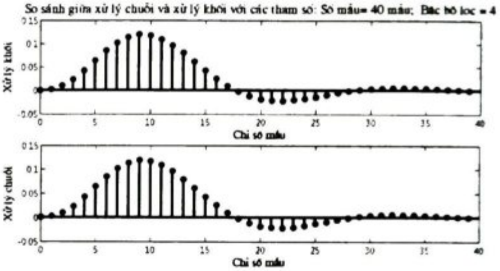
Một ưu điểm của Matlab là dễ dàng tính toán các hệ số bộ lọc a_k và b_k cho rất nhiều nguyên mẫu tương tự khác nhau. Mặc dù, dễ dàng tạo ra các hệ số bộ lọc khi dùng thường trình *filter* trong Matlab, được hướng vào xử lý khối, các bộ lọc IIR có thể được mô phỏng trên cơ sở từng mẫu bằng ba dòng lệnh của Matlab dưới đây:

```
out  = b(1)*in + sreg(1,1)          % Tính đầu ra bộ lọc
sreg = in*b-out*a + sreg;           % Cập nhật các nội dung bộ ghi dịch
sreg = [sreg(1,2:(order + 1)),0];   % Bộ ghi dịch dịch vòng
```

Trong đó a và b được xác định ở ngoài vòng lặp mô phỏng. Tham số *areg* trong đoạn mã trình bày thanh ghi dịch có chiều dài là *order+1* trong đó *order* là bậc của bộ lọc. Ưu điểm của giải pháp này là các thường trình tổng hợp bộ lọc Matlab như *butter, cheby1 và elliptic*, được dùng để tính toán các véc-tơ hệ số bộ lọc a và b. Phải cẩn thận khi dùng để đảm bảo rằng các véc-tơ chứa các hệ số mẫu số và tử số có cùng chiều dài. Nếu hai véc-tơ có chiều dài khác nhau, thì độn thêm số không vào véc-tơ ngắn hơn.

Ví dụ 5.1: Để minh họa kỹ thuật trên, ta xét chương trình Matlab **NVD5_filterex1.m** được cho ở Phụ lục 5A, chương trình xác định đáp ứng xung kim bộ lọc Butterworth bậc 4 sử dụng cả hai quá trình xử lý khối và chuỗi (từng mẫu một).

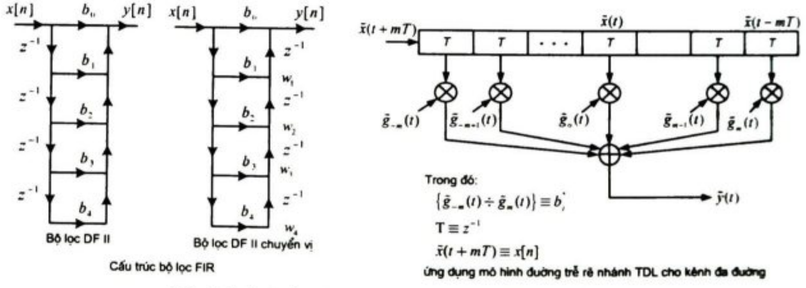
Kết quả chạy chương trình được cho ở hình 5.3. Mô phỏng cho thấy, cả hai kỹ thuật xử lý khối và xử lý nối tiếp đều cho cùng một kết quả.



Hình 5.3: So sánh giữa xử lý chuỗi và xử lý khối

5.3.2. Thực hiện bộ lọc FIR

Chiến lược thực thi các bộ lọc số FIR trực tiếp từ hình 5.2 khi đặt $a_k = 0, k \geq 1$. Hiển nhiên, ta nhận được hai cấu trúc được cho ở hình 5.4. Chúng được coi là cấu trúc đường trễ rẽ nhánh TDL bởi lẽ chúng được thực thi là các dây trễ (các phần tử trễ z-1 được nối tầng) nhân với các trọng số b_k.



Hình 5.4: Cấu trúc bộ lọc FIR và ứng dụng điển hình

5.4. Bộ lọc IIR - Kỹ thuật tổng hợp và đặc điểm

Bộ lọc đáp ứng xung kim vô hạn IIR thường được thiết kế từ nguyên mẫu tương tự (Butterworth, Chebyshev, Elliptic...), và được thực thi bằng các cấu trúc đệ quy. Hệ thống viễn thông hiện đại không dựa vào các nguyên mẫu tương tự như trước đây, ngoại trừ các tầng RF

của hệ thống. Nhiều hệ thống hiện đại như: hệ thống vô tuyến mềm dùng kỹ thuật DSP để xây dựng các khối chức năng (bao gồm các bộ lọc). Trong các hệ thống dựa trên DSP, bài toán triển khai số hóa cho các nguyên mẫu tương tự là không đáng bàn vì các bộ lọc được dùng trong phần cứng vật lý đã thực sự ở dạng số. Kết quả là, kỹ thuật kinh điển để tổng hợp bộ lọc số dùng nguyên mẫu tương tự làm điểm bắt đầu đang mờ nhạt. Tuy nhiên, các thiết kế từ các nguyên mẫu tương tự vẫn được dùng trong nhiều ứng dụng. Một số kỹ thuật thiết kế được hỗ trợ bởi máy tính đã được triển khai cho bộ lọc IIR, một trong số đó được đề cập ở đây. Vì mục đích minh họa các kỹ thuật cơ bản và các lỗi mô phỏng khi dùng các kỹ thuật này, nên ta chỉ xét các ứng dụng rất đơn giản của các kỹ thuật này.

5.4.1. Bộ lọc bất biến xung kim

Giả sử đáp ứng xung *đơn vị* bộ lọc số là đáp ứng xung kim của bộ lọc tương tự được lấy mẫu (bỏ qua tỷ số biên độ). Khi này, bộ lọc số được xem là thực hiện bất biến xung kim của bộ lọc nguyên mẫu tương tự. Nói cách khác, hàm truyền đạt của bộ lọc số $H(z)$ được định nghĩa bởi:

$$H(z) = T\mathcal{Z}\left\{\mathcal{L}^{-1}\left[H_a(s)\right]\Big|_{t=nT}\right\} \tag{5.17}$$

Trong đó $\mathcal{L}$-1(.) là biến đổi Laplace ngược, $\mathcal{Z}$(.) là biến đổi Z, $H_a(s)$ là hàm truyền đạt của bộ lọc tương tự và T là chu kỳ lấy mẫu (là tỷ lệ biên độ).

Để minh họa kỹ thuật bất biến xung kim ta xét nguyên mẫu tương tự đơn giản nhất là:

$$H_a(s) = \frac{a}{s+a} \tag{5.18}$$

Lưu ý rằng: Đây là bộ lọc thông thấp bậc nhất có thành phần một chiều dc ($f = 0$) là 1.

***Ví dụ 5.2**:* Từ (5.17) trước hết biến đổi Laplace ngược của $H_a(s)$ ta được đáp ứng xung kim:

$$\mathcal{L}^{-1}\left[H_a(s)\right] = h_a(t) = a.\mathrm{e}^{-at}.u(t) \tag{5.19}$$

Trong đó $u(t)$ là hàm bước nhảy đơn vị, được dùng để *biểu diễn tính nhân quả của bộ lọc*. Trước khi biến đổi z đáp ứng xung kim phải được lấy mẫu tại các điểm lấy mẫu $t = nT$. Ta được:

$$\mathcal{L}^{-1}\left[H_a(s)\right]\Big|_{t=nT} = h_a[n] = a.\mathrm{e}^{-anT}\,u[n] \tag{5.20}$$

Biến đổi z của $h_a[n]$ là:

$$\mathcal{Z}\left\{\underbrace{\mathcal{L}^{-1}\left[H_a(s)\right]\Big|_{t=nT}}_{h_a[n]}\right\} = \sum_{n=0}^{\infty} a\left[\mathrm{e}^{-aT}\right]^n z^{-n} \tag{5.21}$$

Lấy tổng và nhân với chu kỳ lấy mẫu ta có:

$$H(z) = \frac{aT}{1-\mathrm{e}^{-aT}\,z^{-1}} \tag{5.22}$$

Là tương đương bất biến xung kim của (5.18). Tính cách trong miền tần số của bộ lọc này sẽ được khai thác trong ví dụ 5.5.

5.4.2. Bộ lọc bất biến bước

Bộ lọc bất biến bước thường được dùng trong mô phỏng (lý do được nêu rõ ở phần sau). Với bộ lọc bất biến bước, *đáp ứng xung đơn vị* của bộ lọc số là đáp ứng bước của bộ lọc tương tự được lấy mẫu. Vì đáp ứng bước của bộ lọc tương tự được biểu diễn trong miền tần số là $H_a(s)/s$, nên hàm truyền đạt của bộ lọc bất biến bước là:

$$\underbrace{\frac{1}{1-z^{-1}}H(z)}_{\substack{\text{Đáp ứng của bộ lọc số có}\\ \text{hàm truyền đạt H(z) đối với}\\ \text{bước đơn vị được lấy mẫu}}} = Z\left\{ \underbrace{\left. \underbrace{\mathcal{L}^{-1}\left[\frac{1}{s}H_a(s)\right]}_{\substack{\text{Đáp ứng bước đơn vị của}\\ \text{bộ lọc nguyên mẫu tương tự}}} \right|_{t=nT}}_{\substack{\text{Đáp ứng bước đơn vị của bộ lọc}\\ \text{nguyên mẫu tương tự được lấy mẫu}}} \right\} \tag{5.23}$$

Lưu ý rằng, vế bên trái của (5.23) là đáp ứng của bộ lọc số có hàm truyền đạt *H(z)* đối với bước đơn vị được lấy mẫu. Vế phải của (5.23) biểu diễn biến đổi *z* của đáp ứng bước đơn vị của bộ lọc nguyên mẫu tương tự được lấy mẫu.

Ví dụ 5.3: Trong ví dụ này ta lại sử dụng (5.18) làm nguyên mẫu tương tự. Đáp ứng bước của bộ lọc tương tự là:

$$\frac{1}{s}H_a(s) = \frac{a}{s(s+a)} \tag{5.24}$$

Khai triển ta có:

$$\frac{1}{s}H_a(s) = \frac{1}{s} - \frac{1}{s+a} \tag{5.25}$$

Lấy biến đổi Laplace ngược và lấy mẫu tại $t = nT$ ta có:

$$\mathcal{L}^{-1}\left[\frac{1}{s}H_a(s)\right]\Bigg|_{t=nT} = u[n] - e^{-anT}.u[n] \tag{5.26}$$

Và lấy biến đổi *z* ta có:

$$\frac{1}{1-z^{-1}}H(z) = \sum_{n=0}^{\infty}\left\{1-\left(e^{-aT}\right)^n\right\}z^{-n} \tag{5.27}$$

$$\frac{1}{1-z^{-1}}H(z) = \frac{1}{1-z^{-1}} - \frac{1}{1-e^{-aT}z^{-1}} \tag{5.28}$$

Hàm truyền đạt của bộ lọc số bất biến bước là:

$$H(z) = 1 - \frac{1-z^{-1}}{1-e^{-aT}z^{-1}} \tag{5.29}$$

Viết lại phương trình trên ở dạng chuẩn như được định nghĩa bởi (5.3) ta có:

$$H(z) = \frac{\left[1 - e^{-aT}\right] z^{-1}}{1 - e^{-aT} z^{-1}} \tag{5.30}$$

Các tính cách của bộ lọc này trong miền tần số sẽ được khai thác trong ví dụ 5.5. Kết quả sẽ được so sánh với kết quả bất biến xung kim.

5.4.3. Bộ lọc biến đổi z song tuyến tính

Bộ lọc số biến đổi *z* song tuyến có lẽ được sử dụng nhiều nhất trong mô phỏng bởi lẽ: *(i)* việc tổng hợp bộ lọc sử dụng kỹ thuật biến đổi *z* song tuyến là rất dễ thực hiện và thuần túy số học; *(ii)* bộ lọc biến đổi z song tuyến không tỏ ra chồng phổ.

Kỹ thuật tổng hợp

Kỹ thuật tổng hợp biến đổi z song tuyến ánh xạ nguyên mẫu tương tự $H_a(s)$ thành bộ lọc số bằng biến đổi số học đơn giản. Cụ thể, hàm truyền đạt của bộ lọc số, $H(z)$, được định nghĩa bởi:

$$\underbrace{H(z)}_{\text{Hàm truyền đạt bộ lọc số}} = \underbrace{H_a(s)}_{\text{Hàm truyền đạt nguyên mẫu tương tự}}\Bigg|_{s=C(1-z^{-1})/(1+z^{-1})} \tag{5.31}$$

Trong đó C là hằng số. Kỹ thuật để xác định C sẽ được đề cập ngắn gọn sau. Vì $H_a(s)$ được định nghĩa là tỷ số của các đa thức trong miền s nên ánh xạ số học được định nghĩa bởi (5.31) cho ta $H(z)$ ở dạng tỷ số các đa thức theo z^{-1}. Thực thi bộ lọc cũng như các phần khác đã được đề cập.

Để chứng minh hàm truyền đạt của bộ lọc số có các tính chất mong muốn, ta thay $e^{s_d T}$ (s_d là biến tần số phức đối với bộ lọc số) cho z trong định nghĩa của biến đổi z song tuyến. Ta có:

$$s = \frac{1 - e^{-s_d T}}{1 + e^{-s_d T}} \tag{5.32}$$

Thế $s = j2\pi f_a$ và $s_d = j2\pi f_d$ vào (5.32) ta có:

$$j2\pi f_a = C\frac{e^{j\pi f_d T} - e^{-j\pi f_d T}}{e^{j\pi f_d T} + e^{-j\pi f_d T}} \tag{5.33}$$

Tương đương:

$$2\pi f_a = C\tan\left(\pi f_d T\right) \tag{5.34}$$

$$C = 2\pi f_a \cot\left(\pi f_d T\right) \tag{5.35}$$

Xác định quan hệ giữa đáp ứng tần số của bộ lọc tương tự nguyên mẫu và bộ lọc số. Điều chỉnh C theo cách này sao cho $f_c = f_d$ với tần số lấy mẫu bất kỳ được gọi là làm méo tần trước *(prewarning)*.

Thấy rõ từ các phương trình trên, quan hệ chuyển đổi giữa f_a và f_d là phi tuyến. Trong môi trường mô phỏng thường mong muốn duy trì dạng của hàm truyền đạt cả về biên độ và pha khi ánh xạ bộ lọc tương tự vào bộ lọc số. Việc duy trì dạng hàm truyền đạt cần có quan hệ chuyển đổi giữa f_d và f_a là tuyến tính trong một khoảng càng rộng càng tốt. Vì $\tan(x) \approx x$ khi x nhỏ, biến đổi xấp xỉ tuyến tính yêu cầu $\pi f_d T \leq 1$, hay:

$$f_d \ll \frac{1}{\pi} f_s \tag{5.36}$$

Trong đó f_d là giải tần số xét và f_s là tần số lấy mẫu. Giá trị C sẽ làm cho dễ dàng xác định biến đổi xấp xỉ tuyến tính với $f_d \approx f_a$. Tính tuyến tính yêu cầu:

$$2\pi f_a \approx C(\pi f_d T) \tag{5.37}$$

Nếu ta cần $f_d \approx f_a$ ta có:

$$C = \frac{2}{T} = 2f_s \tag{5.38}$$

Tuy nhiên, việc duy trì $f_d \approx f_a$ với mọi tần số xét như khoảng tần số của dải thông bộ lọc yêu cầu tần số lấy mẫu quá lớn. Dẫn đến thời gian mô phỏng không thực tế.

Ví dụ 5.4: Để minh họa phương pháp tổng hợp biến đổi z song tuyến ta trở lại nguyên mẫu tương tự bậc nhất đơn giản được xác định bởi (5.18). Theo định nghĩa, bộ lọc biến đổi z song tuyến có hàm truyền đạt như sau:

$$H(z) = \left.\frac{a}{s+a}\right|_{s=C\frac{1-z^{-1}}{1+z^{-1}}} \tag{5.39}$$

Dẫn đến:

$$H(z) = \frac{a(1+z^{-1})}{C(1-z^{-1})+a(1+z^{-1})} \tag{5.40}$$

Ở dạng tiêu chuẩn là:

$$H(z) = \frac{\dfrac{a}{C+a} + \dfrac{a}{C+a} z^{-1}}{1 - \dfrac{C-a}{C+a} z^{-1}} \tag{5.41}$$

Tính cách miền tần số của bộ lọc sẽ được khai thác trong ví dụ 5.5.

Ví dụ 5.5: Ví dụ này sẽ so sánh ba bộ lọc số trong ba ví dụ trên. Giả sử tần số lấy mẫu là 100 Hz (T = 0,01), tham số a là $2\pi(10)$ do vậy tần số 3 dB của bộ lọc tương tự là 10 Hz. Với các giá trị giả định này ta có:

$$aT = 0{,}628319 \tag{5.42}$$

$$e^{-aT} = 0{,}533488 \tag{5.43}$$

Theo đó, ta có các hàm truyền đạt cho ba loại bộ lọc số: *(i)* bất biến xung kim; *(ii)* bất biến bước; *(iii)* biến đổi z song tuyến tính (hai phía) nghĩa là:

Hàm truyền đạt của bộ lọc bất biến xung kim có được từ (5.22):

$$H_{ii}(z) = \frac{0,628319}{1-0,533488z^{-1}} \tag{5.44}$$

Hàm truyền đạt của bộ lọc bất biến bước có được từ (5.30):

$$H_{si}(z) = \frac{\left(1-0,533488\right)z^{-1}}{1-0,533488z^{-1}} \tag{5.45}$$

Hàm truyền đạt của bộ lọc biến đổi z hai phía được xác định như sau: Hai phiên bản của bộ lọc biến đổi z song tuyến sẽ được trình bày.

Trước hết, nếu không làm méo tần trước thì đáp ứng biên độ của nguyên mẫu tương tự và bộ lọc số sẽ giống nhau ở các tần số thấp:

$$C = \frac{2}{T} = 200 \tag{5.46}$$

Thế giá trị C và a vào (5.41) ta có:

$$H_{bl}(z) = \frac{0,239057+0,239057z^{-1}}{1-0,5218861z^{-1}} \tag{5.47}$$

Tại đây, giả thiết là giá trị C được lựa chọn sao cho đáp ứng tần số được phù hợp tại tần số 3 dB của bộ lọc tương tự. Khi đó:

$$f_a = f_d = \frac{a}{2\pi} \tag{5.48}$$

Vì vậy, từ (5.35) giá trị C là:

$$C = a\cot\left(\frac{aT}{2}\right) \tag{5.49}$$

Tương đương:

$$C = 40\pi\cot(0,2\pi) = 172,961 \tag{5.50}$$

Sử dụng giá trị C trong (5.41) nhận được:

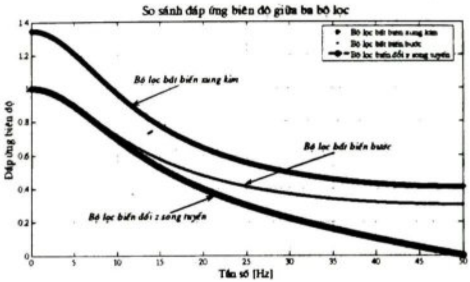
$$H_{bl}(z) = \frac{0,420808+0,420808z^{-1}}{1-0,158384z^{-1}} \tag{5.51}$$

Ta nên so sánh đáp ứng được định nghĩa bởi (5.47) và (5.51).

Mã chương trình Matlab **NVD5_threefilters.m** được cho ở Phụ lục 5A tạo đáp ứng biên độ của các bộ lọc bất biến xung kim, bất biến bước, biến đổi z song tuyến được mô tả bởi (5.47).

Kết quả được minh họa trong hình 5.5. Cần lưu ý: *(i)* bộ lọc bất biến hàm bước và bộ lọc biến đổi z song tuyến đều có cùng độ lợi bằng 1 tại dc. Độ lợi dc của bộ lọc bất biến xung lớn hơn một do chồng phổ; *(ii)* bộ lọc biến đổi z song tuyến có giá trị không tại tần số Nyquist

($f_s/2$) bởi lẽ nguyên mẫu tương tự có giá trị không tại tần số $f = \infty$ và giá trị không này được ánh xạ vào tần số Nyquist.

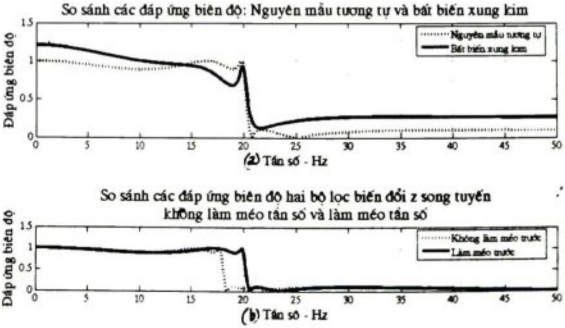


Hình 5.5: So sánh các đáp ứng biên độ

Ví dụ 5.6: Bộ lọc được triển khai trong ví dụ 5.5 dựa vào nguyên mẫu bậc một rất đơn giản và các hệ số bộ lọc: giá trị a_k và b_k được xác định không cần máy tính. Trong ví dụ này ta xét bộ lọc elip bậc 5 có độ nhấp nhô dải thông là 1 dB và suy hao nhỏ nhất của chắn dải là 20 dB. Mặc dù có thể dễ dàng xác định hệ số bộ lọc bằng giải tích ở ví dụ 5.5 nhưng do tính phức tạp của hàm truyền đạt nguyên mẫu tương tự cần nỗ lực đáng kể.

Tuy nhiên Matlab hỗ trợ nhiều kỹ thuật tổng hợp ở dạng hộp công cụ Toolbox tín hiệu và hệ thống. Sẵn có các thường trình để thiết kế bộ lọc số biến đổi z song tuyến và bất biến xung kim. Mã chương trình Matlab tổng hợp bộ lọc elip bậc 5 được cho ở file **NVD5_ellipexam.m** có trong Phụ lục 5A thực hiện bài toán này.

Lưu ý chạy chương trình **NVD5_ellipexam.m** sẽ tạo ra 4 đáp ứng tần số được minh họa ở hình 5.6 trong đó: *(i)* tạo ra đáp ứng biên độ tần số của nguyên mẫu tương tự để so sánh với đáp ứng biên độ tần số của bộ lọc số bất biến xung kim; *(ii)* tạo hai bộ lọc số biến đổi z song tuyến: một có méo tần số và một không méo tần số để minh họa ảnh hưởng của méo tần số.



Hình 5.6: So sánh các đáp ứng biên độ

Để được tường minh, ta so sánh đáp ứng biên độ của nguyên mẫu tương tự, bộ lọc số bất biến xung kim và hai bộ lọc biến đổi z song tuyến được cho ở hình 5.6. Theo đó, hình 5.6(a) so sánh đáp ứng biên độ của bộ lọc số bất biến xung kim với nguyên mẫu tương tự. Lưu ý là ngoài băng thông việc so sánh không được rõ rằng do lỗi chồng phổ, khá rõ trong các bộ lọc elip vì đáp ứng biên độ của nguyên mẫu tương tự không bị giới hạn băng do tính cách dải chắn bộ lọc. Điều này có thể thấy rõ trong đáp ứng biên độ được nâng lên trong cả dải thông và dải chắn của bộ lọc.

Hình 5.6(b) minh họa phương pháp tổng hợp biến đổi z song tuyến tốt hơn nhiều. Nếu méo tần không được dùng thì tần số cắt bị giảm từ 20 Hz đến xấp xỉ 18 Hz. Khi làm méo trước tần số thì tần số cắt của bộ lọc số đạt tới tần số cắt của nguyên mẫu tương tự dẫn đến đáp ứng biên độ gần với đáp ứng biên độ của nguyên mẫu trong hầu hêt dải tần. Tuy nhiên, có một điểm khác biệt. Tại tần số Nyquist ($f_s/2 = 50Hz$) đáp ứng tần số của nguyên mẫu là 0,1 (tương ứng với suy hao 20 dB) trong khi đó đáp ứng biên độ của bộ lọc số biến đổi z song tuyến là 0 (tương ứng suy hao là ∞ dB). Bộ lọc tương tự có giá trị không tại tần số $f = \infty$ còn với biến đổi z song tuyến điểm không này được ánh xạ vào tần số Nyquist.

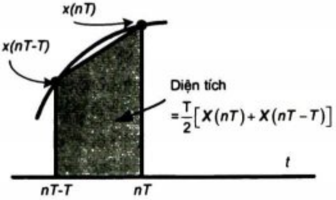
Trường hợp đặc biệt: Tích phân hình thang

Thuật toán tích phân quan trọng thường được dùng trong mô phỏng là thuật toán tích phân hình thang. Tổng quát, tích phân thời gian rời rạc được cho bởi biểu thức:

$$y[n] = y[n-1] + \Delta(n-1,n) \tag{5.52}$$

$y[n]$ và $y[n-1]$ là đầu ra bộ tích phân tại chỉ số n và n - 1, và $\Delta(n-1,n)$ là diện tích gia lượng cộng thêm vào tích phân theo gia số thời gian $(n-1)T < t \le nT$. Diện tích gia lượng được minh họa ở hình 5.7 và được cho bởi:

$$\Delta\left(n-1,n\right) = \frac{T}{2}\left(x[n] + x[n-1]\right) \tag{5.53}$$



Hình 5.7: Tích phân hình thang

Theo đó:

$$y[n] - y[n-1] = \frac{T}{2}\left(x[n] + x[n-1]\right) \tag{5.54}$$

Lấy biến đổi z hai phía (5.54) nhận được hàm truyền đạt của tích phân hình thang. Kết quả là:

$$H(z)=\frac{Y(z)}{X(z)}=\frac{T}{2}\times\frac{1+z^{-1}}{1-z^{-1}} \tag{5.55}$$

Nếu ta thiết kế thuật toán tích phân dựa trên kỹ thuật tổng hợp biến bởi z song tuyến, kết quả là:

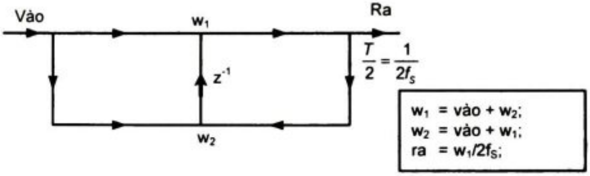
$$H(z)=H_a(s)\Big|_{s=\frac{2}{T}\times\frac{1-z^{-1}}{1+z^{-1}}} \tag{5.56}$$

Vì tích phân trong miền thời gian là tương đương với chia cho s trong miền s nên:

$$H(z)=\frac{1}{s}\Big|_{s=\frac{2}{T}\times\frac{1-z^{-1}}{1+z^{-1}}}=\frac{T}{2}\times\frac{1+z^{-1}}{1-z^{-1}} \tag{5.57}$$

Cũng giống như tích phân hình thang được biểu diễn bởi (5.55)

Đồ hình tín hiệu tương ứng để thực hiện DF II chuyển vị của bộ tích phân hình thang được minh họa ở hình 5.8. Lưu ý rằng, ngoại trừ lấy tỷ số biên độ là $T/2$ tất cả là đơn vị. Tích phân hình thang chỉ là một trong rất nhiều thuật toán được sử dụng trong các chương trình mô phỏng.



Hình 5.8: Đồ hình dòng tín hiệu cho bộ tích phân hình thang và mã Matlab

5.4.4. Thiết kế bộ lọc số IIR với sự hỗ trợ của máy tính

Vài thập niên gần đây đã triển khai nhiều kỹ thuật thiết kế bộ lọc số FIR và IIR với sự hỗ trợ của máy tính. Kỹ thuật FIR nhận được nhiều ứng dụng hơn kỹ thuật IIR nhưng kỹ thuật IIR đôi khi được sử dụng do cấu trúc nhỏ gọn (ít thành phần trễ hơn). Hai ưu điểm nổi trội thiết kế với sự hỗ trợ bởi máy tính là: *(i)* có thể thiết kế các bộ lọc số mong muốn mà không cần một nguyên mẫu tương tự; *(ii)* cho phép đạt được dung hòa, chẳng hạn giữa mức độ phức tạp (số lượng hệ số bộ lọc) và mức độ chính xác (lỗi giữa đáp ứng biên độ mong muốn và thực tế).

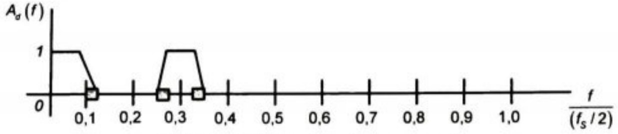
Kỹ thuật phổ biến để triển khai bộ lọc IIR là điều chỉnh hệ số bộ lọc cho đến khi đạt được đáp ứng mong muốn, phổ cập nhất là xác định đặc điểm bộ lọc mong muốn, xác định đáp ứng biên độ và bậc N của bộ lọc. Dẫn đến bài toán tối ưu bị hạn chế do bậc N là giới hạn.

Ví dụ 5.7: Ta triển khai bộ lọc định kênh để có được các tính chất như trong hình 5.9. Lưu ý: *(i)* bộ lọc cho hai kênh qua nó, một kênh được dịch về dc; *(ii)* tần số trong hình 5.9 được chuẩn hóa theo tần số Nyquist (một nửa tần số lấy mẫu). Dưới dạng tần số chuẩn hóa này $f_N=f/(f_s/2)$, đáp ứng biên độ của bộ lọc được định nghĩa là:

$$A_d(f)=\begin{cases}1, & |f|<0{,}1 \quad 0{,}25<f<0{,}35\\ 0, & 0{,}12<f<0{,}23 \quad 0{,}37<f<1\end{cases} \tag{5.58}$$

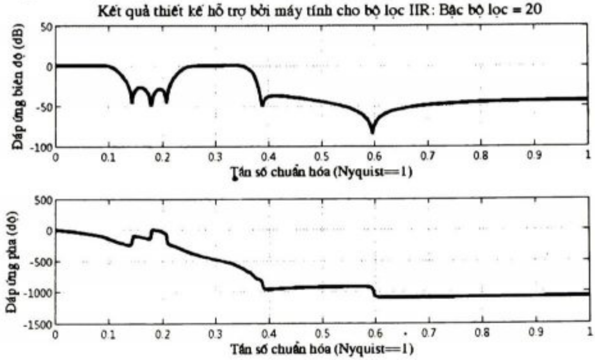
Ba băng tần không có trong biểu thức của $A_d(f)$ là các băng tần dịch, được ký hiệu vùng đen trong hình 5.9.

Mã chương trình Matlab để thiết kế bộ lọc này được cho ở file **NVD5_yw.m** trong Phụ lục 5A.



Hình 5.9: Đáp ứng biên độ mong muốn

Lưu ý rằng, nếu bộ lọc có bậc 20 được tạo ra thì cả tử số và mẫu số của đa thức *H(z)* là đa thức bậc 20 của z^{-1}. Chạy chương trình mô phỏng Matlab kết quả được cho ở hình 5.10, suy hao dải chắn là 30 dB hoặc tốt hơn. Dải thông được định dạng khá hợp lý và đáp ứng pha xấp xỉ tuyến tính trong dải thông. Tăng bậc bộ lọc sẽ cải thiện lỗi xấp xỉ nhưng lại tăng tính phức tạp hay tăng thời gian mô phỏng.



Hình 5.10: Kết quả thiết kế IIR với sự hỗ trợ của máy tính

5.4.5. Nguồn lỗi trong bộ lọc IIR

Thấy rõ, nguồn lỗi phụ thuộc vào phương pháp tổng hợp được dùng. Đa phần các lỗi này được giảm thiểu bằng cách sử dụng tần số lấy mẫu rất lớn trong mô phỏng nhưng thời gian mô phỏng lại quá lâu đôi khi không thực sự cần thiết. Theo đó, việc chọn tần số lấy mẫu phải cân nhắc giữa tính chính xác và thời gian mô phỏng. Nguồn lỗi được tóm tắt trong bảng 5.1.

Bảng 5.1: Các nguồn lỗi bộ lọc số IIR

Kỹ thuật tổng hợp	Nguồn lỗi	Giảm thiểu lỗi
Bất biến xung kim	Chồng phổ	Chọn tần số lấy mẫu cao hơn
Bất biến bước	Chồng phổ	Chọn tần số lấy mẫu cao hơn

Biến đổi Z song tuyến (2 phía)	Méo tần số	Chọn một tần số lấy mẫu cao hơn nhiều tần số tới hạn cao nhất
Phương pháp CAD	Lỗi lấy xấp xỉ	Tăng bậc của bộ lọc hoặc dùng phương pháp tổng hợp khác phù hợp với ứng dụng hơn

5.5. Bộ lọc FIR - Kỹ thuật tổng hợp và đặc điểm

Nếu như đáp ứng xung kim của bộ lọc $h[n]$ là hữu hạn, hay nó được lấy hữu hạn bằng cách cắt bớt đáp ứng xung kim vô hạn thì đầu ra của bộ lọc trong miền thời gian rời rạc được cho bởi:

$$y(nT) = T\sum_{k=0}^{N} h(kT).x\{(n-k)T\} \tag{5.59}$$

Được viết theo DSP chuẩn là:

$$y[n] = \sum_{k=0}^{N} h[n]x[n-k] = \sum_{k=0}^{N} b_k x[n-k] \tag{5.60}$$

Hoặc ở dạng biến đổi z là:

$$Y(z) = \sum_{k=0}^{N} b_k z^{-k} X(z) \tag{5.61}$$

Phương trình (5.60) định nghĩa bộ lọc số FIR, là phiên bản rời rạc theo thời gian của tích chập đối với bộ lọc tương tự. Lưu ý rằng, (5.60) và (5.61) giống với mô hình bộ lọc được định nghĩa bởi (5.1) và (5.3) với $a_k = 0$ khi $k \geq 1$. Bộ lọc là FIR bởi lẽ trong $h[n]$ hầu như có $N + 1$ thành phần khác không. Vì $a_k = 0$ khi $k \geq 1$ nên bộ lọc FIR không có đường hồi tiếp. Phép toán tổng chập trong miền thời gian được định nghĩa trong (5.60), được mô phỏng bằng hàm *filter* trong Matlab. Nếu ta nghiên cứu hàm *filter* bằng lệnh *doc filter* từ cửa sổ làm việc của Matlab, khi này ta sẽ nhận được trình bày ngắn gọn và súc tích về định nghĩa hàm cũng như cách sử dụng và *rất dễ hiểu* thông qua các minh họa ở đó.

Bộ lọc FIR là rất hấp dẫn vì các lý do cơ bản sau

1. Không phải tất cả các bộ lọc trong viễn thông đều có thể được biểu diễn dưới dạng hàm truyền đạt trong miền biến đổi Laplace do vậy kỹ thuật IIR được mô tả trong phần trước là không thể trực tiếp áp dụng được. Hai bộ lọc quan trọng thuộc loại này là bộ lọc định dạng xung SQRC, và bộ lọc Doppler Jakes. Dễ dàng mô phỏng chúng bằng giải pháp FIR.

2. Trong nhiều mô phỏng, số liệu bộ lọc được cho ở dạng đáp ứng tần số đo đạc hoặc số liệu của đáp ứng xung kim. Rất dễ thực hiện mô phỏng các bộ lọc này bằng giải pháp FIR. Trong khi sẵn có một số kỹ thuật để làm phù hợp mô hình ARMA với số liệu đáp ứng tần số thì việc lấy xấp xỉ như vậy không mang lại kết quả mong muốn khi số liệu đáp ứng tần số là không liên tục.

3. Bằng giải pháp FIR, ta có thể xác định đáp ứng pha và biên độ tùy ý, và chúng có thể độc lập với nhau.

4. Do bộ lọc FIR không có hồi tiếp nên luôn luôn ổn định.

Mô hình mô phỏng FIR có một nhược điểm lớn là hiệu quả tính toán không bằng bộ lọc IIR. Mỗi mẫu được tạo ra theo (5.60) cần phải có N phép nhân và phép cộng. Nếu đáp ứng xung rất dài (N > 1024 điểm), thì thuật toán FIR sẽ chạy lâu hơn rất nhiều so với các thuật toán IIR thông thường. Với bộ lọc IIR số lượng các phép cộng và nhân để tạo ra một mẫu được xác định bởi bậc của bộ lọc chứ không phải là chiều dài của đáp ứng xung kim như trường hợp FIR.

Hiệu quả tính toán của mô hình FIR có thể được cải thiện bằng việc dùng DFT/FFT để tính tổng chập. Khi sử dụng DFT/FFT cần đặc biệt các lưu ý sau:

1. DFT/FFT là tuần hoàn và tạo các đầu ra chập một cách tuần hoàn hoặc vòng. Để có được các đầu ra chập tuyến tính thì phải độn các số 0 vào chuỗi đầu vào và chuỗi đáp ứng xung kim. Để có được hiệu quả tính toán cao nhất thì chiều dài của các véc tơ đã được độn phải là lũy thừa của 2 do đó tận dụng thuật toán FFT cơ số 2.

2. Nếu chuỗi đầu vào quá dài thì trước hết phân thành các khối nhỏ không chồng lấn, sau đó thực hiện tích chập cho mỗi khối đầu vào để tìm khối đầu ra tương ứng, cuối cùng xếp chồng lần lượt và cộng để tạo ra chuỗi đầu ra. Nếu khối đầu vào ngắn (nhỏ hơn 216 mẫu) thì khối đầu ra sẽ được tính toán trong một lần. Phương pháp xếp chồng và cộng của thuật toán DFT/FFT dựa trên tổng chập được thực hiện bằng hàm Matlab có sẵn **fftfit**. Ta có thể tìm hiểu hàm **fftfit** bằng lệnh **doc fftfit.**

3. Tổng chập dựa trên DFT/FFT là một quá trình xử lý khối, mẫu đầu tiên của khối đầu ra chỉ được tạo ra sau khi tích lũy đủ số mẫu đầu vào để thực hiện DFT/FFT. Do trễ này mà bộ lọc DFT/FFT thường không được sử dụng để mô phỏng bộ lọc (mà bộ lọc là một phần của vòng lặp hồi tiếp).

Thấy rõ, hiệu quả tính toán tổng chập trong miền thời gian dựa trên DFT là tỉ lệ với $\frac{\log_2 N}{N}$. Nếu chiều dài của đáp ứng xung kim nhỏ hơn 128 mẫu, sẽ không có nhiều khác biệt về khối lượng tính toán giữa tổng chập trong miền thời gian và thực hiện bằng DFT/FFT.

Hiệu quả tính toán của phương pháp FIR (trong miền thời gian và DFT/FFT dựa trên tích chập) được cải thiện nếu chiều dài của đáp ứng xung kim được rút ngắn. Việc cắt giảm cũng tương đương với việc nhân đáp ứng xung kim $h[k]$, $k = 0, 1, 2, 3,...$ với một hàm cửa sổ $w[k]$ có chu kỳ là N mẫu. Giá trị N được chọn sao cho ít nhất 98% năng lượng của $h[n]$ được chứa trong cửa sổ. Nói cách khác:

$$\sum_{k=0}^{N} |h[k]|^2 = 0,98\sum_{k=0}^{\infty} |h[k]|^2 \qquad (5.62)$$

Hàm của sổ đơn giản nhất là hàm cửa sổ chữ nhật nó sẽ loại bỏ các mẫu đáp ứng xung kim ứng với k > N. Nói cách khác:

$$w[k] = \begin{cases} 1, & 0 \le k \le N \\ 0, & \text{nếu khác} \end{cases} \qquad (5.63)$$

Tồn tại rất nhiều hàm cửa sổ khác được sử dụng.

Việc cắt ngắn (cửa sổ hóa) trong miền thời gian tương đương với tổng chập trong miền tần số, nghĩa là $H_T(f) = H(f) \otimes W(f)$ trong đó $\otimes$ là tổng chập. Một cách lý tưởng hàm cửa sổ

trong miền tần số là $W(f) = \delta(f)$ vì khi lấy tích chập với một xung kim sẽ không làm thay đổi $H(f)$. *Một xung kim trong miền tần số sẽ tương ứng với một hằng số trong miền thời gian nên nó có chu kỳ là vô hạn chứ không phải là hữu hạn. Theo đó, nếu đặc tính lý tưởng trong miền tần số thì tương ứng không lý tưởng trong miền thời gian và ngược lại.* Vì vậy, khi lựa chọn hàm cửa sổ cần phải cân nhắc cẩn thận và tính đến méo do việc cửa sổ hóa. Có thể giảm thiểu méo này bằng cách chọn hàm cửa sổ phù hợp. Các đặc tính mong muốn của hàm của sổ gồm (luôn lưu ý rằng, về mặt lý tưởng luôn mong muốn có xung kim trong miền tần số).

1. "Búp phổ chính hẹp" trong miền tần số chứa đựng hầu hết năng lượng
2. Các búp phụ nhỏ.

Hàm cửa sổ được sử dụng phổ biến nhất là hàm cửa sổ chữ nhật, cửa sổ Hamming và cửa sổ Kaiser. Khi chọn hàm cửa sổ, cần phải dung hoa giữa việc giảm thiểu méo tín hiệu với khối lượng tính toán.

5.5.1. Thiết kế từ đáp ứng biên độ

Kỹ thuật cơ bản để thiết kế bộ lọc số FIR dựa trên đáp ứng tần số (biên độ và pha) và đáp ứng xung kim đơn vị của bộ lọc, chúng là cặp biến đổi Fourier. Ta rút ra đáp ứng xung kim đơn vị bằng cách xác định rõ đáp ứng biên độ mong muốn $A(f)$ sau đó biến đổi Fourier ngược. Đáp ứng biên độ đích thường được xác định là thực chẩn vì vậy đáp ứng xung kim đơn vị cũng là thực và chẩn. Vì đáp ứng xung kim là chẩn (không nhân quả) và để thực thi hệ thống trong miền thời gian thì đáp ứng xung kim phải được cắt ngắn sao cho hữu hạn và phải được dịch thời để nó nhân quả. Việc cắt xén đáp ứng xung kim phải được thực hiện cẩn thận nếu không nó sẽ gây ra lỗi rất lớn. Lựa chọn cửa sổ phù hợp sẽ giảm ảnh hưởng các lỗi này. Dịch thời xung kim làm cho bộ lọc đáp ứng pha tuyến tính tương đương với trễ nhóm bằng dịch thời. Kỹ thuật này tạo ra bộ lọc có đáp ứng biên độ tùy ý và dịch pha tuyến tính. Nếu muốn có bộ lọc có cả đáp ứng biên độ và pha chiến lược thì phải xác định rõ hàm truyền đạt phức.

Đáp ứng biên độ của bộ lọc số là hàm liên tục của biến tần số *f*, tuần hoàn theo tần số lấy mẫu và được biểu diễn dưới dạng chuỗi Fourier. Các hệ số Fourier là rời rạc, tạo ra đáp ứng xung kim bộ lọc số mong muốn. Ta sẽ thấy trong các ví dụ dưới đây, tìm đáp ứng xung kim bằng cách biến đổi Fourier ngược lên đáp ứng tần số hoặc thực hiện IFFT lên tập các mẫu của đáp ứng biên độ. Kỹ thuật này rất cơ bản và nếu áp dụng cẩn thận nó sẽ tạo ra các bộ lọc rất tốt.

Vì đáp ứng biên độ của bộ lọc số là tuần hoàn theo tần số lấy mẫu nên nó được khai triển ở dạng chuỗi Fourier:

$$H(e^{j2\pi fT}) = \sum_{n=0}^{M-1} h[n] e^{-j2\pi nfT} \tag{5.64}$$

Trong đó $h[n]$ là hệ số Fourier, M là chiều dài của đáp ứng xung kim và $T = 1/f_s$. Lưu ý rằng, (5.64) chính xác là (5.6) và (5.4) được sử dụng để đạt được đáp ứng tần số trạng thái ổn định từ hàm truyền đạt $H(z)$. Giả sử M = 2L+1 (lý do sẽ đề cập sau). Với giả thiết này ta thế vào và thay đổi hệ số $k = n - L$ ta có:

$$H(e^{j2\pi fT}) = \sum_{k=-L}^{L} h[k+L] e^{-j2\pi (k+L) fT} \tag{5.65}$$

$$H(e^{j2\pi fT}) = e^{-j2\pi LfT}\sum_{k=-L}^{L} h[k+L].e^{-j2\pi kfT} \tag{5.66}$$

Được viết dưới dạng:

$$H(e^{2j\pi fT}) = e^{-j2\pi LfT}.H_1(e^{j2\pi fT}) \tag{5.67}$$

Trong đó:

$$H_1(e^{j2\pi fT}) = \sum_{k=-L}^{L} h_1[k]e^{-j2\pi kfT} \tag{5.68}$$

Rõ ràng $h_1[k] = h[k+L]$ là đáp ứng xung kim của bộ lọc nhân quả, được định nghĩa bởi (5.64) được dịch L mẫu. Lưu ý, bộ lọc được định nghĩa bởi (5.68) không phải là bộ lọc nhân quả. Tuy nhiên, nó rất dễ để thiết kế bằng kỹ thuật Fourier cơ bản và có thể chuyển thành nhân quả bằng cách dịch thời đáp ứng xung kim phù hợp. Đáp ứng biên độ của các bộ lọc được định nghĩa bởi $h[n]$ và $h_1[n]$ là giống nhau và hàm truyền đạt chỉ khác nhau bởi dịch pha tuyến tính như được định nghĩa bởi (5.67). Vì vậy, quy trình thiết kế dựa vào (5.68).

Giả sử bộ lọc có đáp ứng biên độ cho trước là $H_1(e^{j2\pi fT}) = A(f)$. Nhân cả hai vế của (5.68) với $\exp(j2\pi mfT)$ ta có:

$$A(f).e^{j2\pi mfT} = \sum_{k=-L}^{L} h_1[k]e^{j2\pi(m-k)fT} \tag{5.69}$$

Tích phân hai phía của (5.69) trên độ rộng băng tần mô phỏng, nó là một chu kỳ của $H_1(e^{j2\pi fT})$ ta có:

$$\sum_{k=-L}^{L} h_1[k].I(m,k) = \int_{-f_s/2}^{f_s/2} A(f)e^{j2\pi mfT}\,df \tag{5.70}$$

Trong đó:

$$I(m,k) = \int_{-f_s/2}^{f_s/2} e^{j2\pi(m-k)fT}\,df \tag{5.71}$$

Tích phân, và thừa nhận $f_sT = 1$ ta được:

$$I(m,k) = \frac{1}{T}\frac{\sin\pi(m-k)}{\pi(m-k)} = \frac{1}{T}\delta(m-k) \tag{5.72}$$

Thế kết quả này vào (5.70) ta có:

$$h_1[m] = T\int_{-f_s/2}^{f_s/2} A(f)e^{j2\pi mfT}\,df, \qquad -L \le m \le L \tag{5.73}$$

Nó là phương trình thiết kế cơ bản của ta.

Ví dụ 5.8: Thiết kế bộ lọc thông thấp số, xấp xỉ với bộ lọc số lý tưởng có độ rộng băng tần là $\lambda f_N = \lambda f_s/2$, trong đó f_N là tần số Nyquist, f_s là tần số lấy mẫu, λ là tham số nằm giữa 0 và 1. Vì vậy, đáp ứng biên độ mong muốn là:

$$A(f)=\begin{cases}1, & |f|<\lambda f_s/2\\ 0, & |f|\geq \lambda f_s/2\end{cases} \quad (5.74)$$

Từ (5.73) ta có:

$$h_1[m]=T\int_{-\lambda f_s/2}^{\lambda f_s/2}(1)e^{j2\pi mfT}\,df \quad (5.75)$$

Thực hiện tích phân trên ta có:

$$h_1[m]=T.\frac{1}{\pi mT}.\frac{1}{j2}\left[e^{j\pi m\lambda f_sT}-e^{-j\pi m\lambda f_sT}\right] \quad (5.76)$$

Vì $f_sT=1$ nên (5.76) được viết:

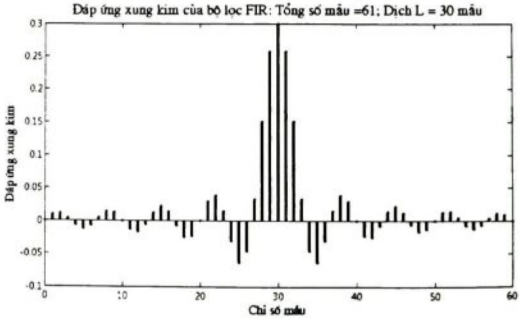
$$h_1[m]=\frac{1}{\pi m}\sin\left(\pi m\lambda\right) \quad (5.77)$$

Lưu ý là $h_1[0]=h[L]=\lambda$.

Mã chương trình Matlab để khai thác kỹ thuật này được cho ở file **NVD5_FIRdesign.m** trong Phụ lục 5A.

Khi nghiên cứu chương trình mô phỏng cần lưu ý: *(i)* trong chương trình này sử dụng L = 30 và λ = 0,3; *(ii)* **eps** được thêm vào chỉ số *m* để tránh tình trạng không xác định tại *m* = 0; *(iii)* Cần phải dịch *L* = 30 mẫu để làm cho xung kim là nhân quả được thực hiện theo cách Matlab đánh chỉ số véc tơ.

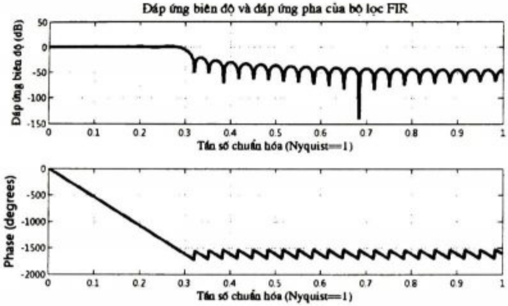
Đáp ứng xung kim được tạo ra bởi chương trình Matlab, kết quả được minh họa ở hình 5.11. Cần lưu ý: *(i)* $h[30]=0,3$ đã được xác định trước; *(ii)* xung kim là hàm chẵn qua trọng số trung tâm tại $m=L=30$. Để được tường minh, cần đọc kỹ mã chương trình Matlab, chạy từng lệnh và thay đổi các tham số cho mỗi lần chạy chương trình.



Hình 5.11: Đáp ứng xung kim của bộ lọc FIR

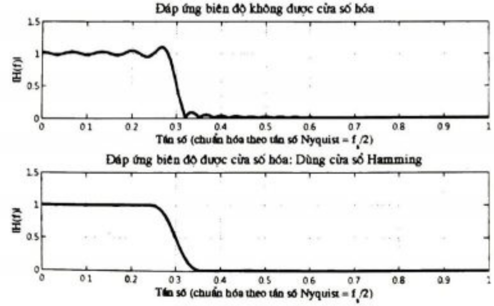
Đáp ứng pha và biên độ của bộ lọc FIR, được tạo ra bởi lệnh **freqz** trong chế độ vẽ mặc định (để được sáng tỏ, ta tìm hiểu hàm **freqz** dùng lệnh **doc freqz**), kết quả được minh họa ở

hình 5.12. Như mong đợi, bộ lọc có pha tuyến tính vì đáp ứng xung kim là hàm chẵn qua trọng số trung tâm (xem hình 5.11). Dịch pha tuyến tính (trễ nhóm không đổi) do dịch đáp ứng xung kim một khoảng L mẫu làm cho bộ lọc có tính nhân quả (xem hình và mã chương trình Matlab). Đáp ứng pha tựa hình răng cưa trong băng chặn do đổi dấu của đáp ứng biên độ (xem hình 5.12).



Hình 5.12: Đáp ứng biên độ và pha của bộ lọc

Đáp ứng biên độ trong băng thông của bộ lọc là phẳng ở hình 5.12. Để khảo sát và trực quan hóa mô phỏng, vẽ đáp ứng biên độ lần thứ hai dùng lấy tỉ lệ tuyến tính để tránh nén biên độ do lấy tỉ lệ loga. Kết quả được cho ở nửa trên của hình 5.13. Cho thấy rằng có nhiều gợn sóng trong hình là do cắt ngắn bớt đáp ứng xung kim bằng cách nhân đáp ứng xung kim với hàm cửa sổ, tương đương với lấy tích chập đáp ứng tần số lý tưởng với $\sin(\beta f)/\beta f$ trong miền tần số (Giá trị β được xác định bởi độ rộng cửa sổ). Có thể giảm độ gợn sóng do cửa sổ hóa bằng cách dùng hàm cửa sổ mịn hơn chuyển dịch từ $w[n]=1$ tới $w[n]=0$. Nhắc lại là ta mong muốn hàm cửa sổ giống hàm xung kim hơn nữa trong miền tần số.



Hình 5.13: Ảnh hưởng của cửa sổ hóa

Để được tường minh, ta chạy chương trình từng lệnh và thay đổi các giá trị khác nhau mỗi khi chạy chương trình cũng như tìm hiểu các hàm trong mã chương trình.

Hàm cửa sổ thường dùng là cửa sổ Hamming được định nghĩa bởi các trọng số:

$$w[n] = 0,54 + 0,46\cos\left(\frac{\pi n}{L}\right), \quad -L \le n \le L \qquad (5.78)$$

Lưu ý rằng, hàm cửa sổ $w[n]$ phải được dịch thời L mẫu sao cho $w[0]$ được trung tâm trên $h[n]$. Kết quả sử dụng bộ lọc Hamming được minh họa ở nửa dưới của hình 5.13. Lưu ý rằng, phần gợn sóng cho cả dải thông và dải chắn đều được nén khi sử dụng của sổ Hamming.

Ví dụ 5.9: Thiết kế bộ lọc số thông dải có đáp ứng biên độ của bộ lọc tương tự Butterworth pha tuyến tính. Bộ lọc tương tự Butterworth được định nghĩa bởi đáp ứng biên độ:

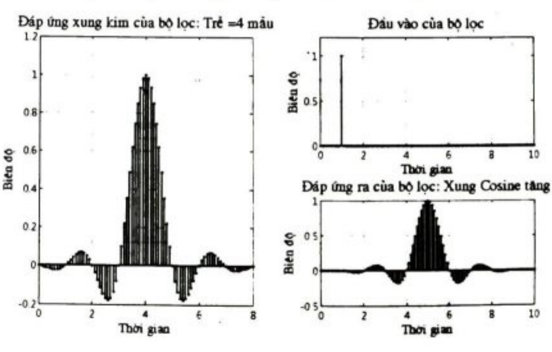
$$A(f) = \frac{1}{\sqrt{1+(f/f_c)^n}} \qquad (5.79)$$

Trong đó f_c là độ rộng băng tần, hay tần số 3dB và n là bậc của bộ lọc. Tiếp theo là lấy mẫu đáp ứng tần số, các mẫu đáp ứng tần số được cho bởi:

$$A(f_k) = \frac{1}{\sqrt{1+(f_k/f_c)^n}} \qquad (5.80)$$

Thực hiện thuật toán IFFT lên các mẫu tần số $A\left(f_k\right)$ sẽ tạo ra đáp ứng xung kim. Cần lưu ý rằng, khi thực hiện IFFT N điểm, các mẫu thời gian âm sẽ xuất hiện trong véc-tơ kết quả có dải chỉ số từ $(N/2)+1$ đến N và phải được đánh chỉ số lại để có đáp ứng xung kim. Mã chương trình Matlab tính toán các mẫu đáp ứng xung kim cho bộ lọc pha tuyến tính có đáp ứng biên độ Butterworth là file **NVD5_firbutter.m** được cho ở Phụ lục 5A.

Kết quả chạy chương trình Matlab tạo ra đáp ứng xung kim được minh họa ở hình 5.14 cho bậc = 1 và bậc = 30. Đáp ứng là một hàm mũ giảm hai phía (hai phía do đáp ứng biên độ $A(f)$ là thực). Khi bậc là 30, thì đáp ứng xung kim sẽ xấp xỉ hàm **sinc** vì bộ lọc Butterworth có bậc rất cao xấp xỉ bộ lọc lý tưởng. Do vậy, kết quả được minh họa ở hình 5.14 là đúng.



Hình 5.14: Kết quả tổng hợp bộ lọc - đáp ứng xung kim

Thực tế, trong mô phỏng mong muốn giảm thiểu khối lượng tính toán bằng cách giảm thiểu số lượng trọng số (thành phần h_k được dùng để biểu diễn đáp ứng xung kim). Khi giảm thiểu số lượng trọng số sẽ giảm thiểu số phép nhân và số lượng các tầng trong đường trễ rẽ nhánh (thanh ghi dịch). Xuất hiện nhiều trọng số trong đáp ứng xung kim bậc = 1 là xấp xỉ bằng không và có thể được khử với mức độ ảnh hưởng không đáng kể lên đáp ứng biên độ. Điều này cũng đúng với đáp ứng xung kim bậc 30. Vì vậy, ta nên thực hiện cửa sổ hóa đáp ứng xung kim bằng hàm cửa số chữ nhật có độ rộng đủ lớn để chứa tất cả các thành phần quan trọng trong đáp ứng xung kim đó và loại bỏ các thành phần ngoài cửa sổ.

5.5.2. Thiết kế từ đáp ứng xung kim

Trong ví dụ 5.9, bộ lọc số được tổng hợp từ chỉ tiêu đặc tính đáp ứng biên độ và pha, là một ứng dụng của *lấy mẫu tần số*. Trong nhiều ứng dụng, biểu thức giải tích đáp ứng xung kim của bộ lọc số. Khi này, lấy mẫu đáp ứng xung kim cho ta thiết kế bộ lọc FIR. Thường được dùng để định dạng xung trong truyền dẫn tín hiệu và số liệu là chuỗi xung kim được phân tách về thời gian một khoảng T. Trường hợp nhị phân ta có:

$$d(t) = \sum_k d_k \delta(t - kT)$$

$$\text{trong đó } d_k = \begin{cases} 1, & \text{bit nhị phân } 1 \\ -1, & \text{bit nhị phân } 0 \end{cases} \tag{5.81}$$

Cho nó qua bộ lọc định dạng xung có đáp ứng xung kim là $p(t)$ ta được dạng sóng:

$$x(t) = \sum_k d_k p(t - kT) \tag{5.82}$$

Sau đây sẽ xét cho hai trường hợp thông dụng.

***Ví dụ 5.10*:** Định dạng xung $p(t)$ được dùng đối với truyền số liệu thường được chọn là một dạng xung thỏa mãn tính chất Nyquist ISI = 0. Một ví dụ về định dạng xung ISI = 0 là xung cosin tăng được cho trong miền tần số là:

$$P(f) = \begin{cases} T, & 0 \le |f| \le \dfrac{1-\beta}{2T} \\ \dfrac{T}{2}\left[1 + \cos\dfrac{\pi T}{\beta}\left(|f| - \dfrac{1-\beta}{2T}\right)\right], & \dfrac{1-\beta}{2T} < |f| \le \dfrac{1+\beta}{2T} \\ 0, & |f| > \dfrac{1-\beta}{2T} \end{cases} \tag{5.83}$$

T là khoảng thời gian xung hay thời gian ký hiệu. Lấy biến đổi Fourier ngược ta được dạng xung:

$$p(t) = \frac{\sin \pi t/T}{\pi t/T} \frac{\cos(\pi\beta t/T)}{1 - 4\beta^2 t^2/T^2} \tag{5.84}$$

Thấy rõ, đây là một xung không nhân quả. Cụ thể, một khi làm trễ xung đi một số nguyên lần chu kỳ ký hiệu, giả sử là mT và cắt ngắn xung còn $2mT$. Giá trị m để dung hòa giữa

các yêu cầu độ chính xác và tính tiện dụng. Sau đó ta lấy k mẫu trên một chu kỳ ký hiệu sao cho $T = kT_s$ với T_s là chu kỳ lấy mẫu. Thay t bằng $t\text{-}t_d = t\text{-}mT$, đặt $t = nT_S$ và $T = kT_S$ nhận được:

$$\frac{t}{T} \rightarrow \frac{nT_s - mkT_s}{kT_s} = \frac{n}{k} - m \tag{5.85}$$

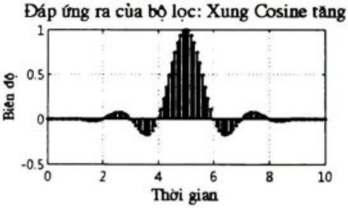
Thế vào trong công thức (5.84) cho ta chuỗi mẫu $p[n]$ biểu diễn đáp ứng xung kim của bộ lọc số là:

$$p[n] = \frac{\sin \pi \left\{\frac{n}{k} - m\right\}}{\pi \left\{\frac{n}{k} - m\right\}} \times \frac{\cos\left(\pi\beta\left\{\frac{n}{k} - m\right\}\right)}{1 - 4\beta^2\left\{\frac{n}{k} - m\right\}^2}, \quad 0 \le n \le 2m \tag{5.86}$$

Để minh họa đáp ứng cosin tăng, ta đặt $x[n] = \delta[n-1]$ là đầu vào của bộ lọc có đáp ứng xung kim $p[n]$. Đầu ra là:

$$y[n] = p[n] \otimes x[n] = p[n-1] \tag{5.87}$$

Kết quả được minh họa ở hình 5.15 với các tham số $\beta = 0{,}32$, $k = 10$ mẫu trên một ký hiệu và $m = 4$ ký hiệu. Thấy rõ, $p[n]$ bằng không tại các vị trí là bội số nguyên lần khoảng thời gian xung giả định $T = 1$. Mã chương trình Matlab được cho bởi file **NVD5_rcossim.m** để tạo ra hình 5.15 trong Phụ lục 5A.



Hình 5.15: Minh họa xung cosin tăng $\beta = 0{,}32$, $k = 10$ mẫu trên một ký hiệu và m = 4 ký hiệu

Trong nhiều thiết kế hệ thống, thực thi hàm truyền đạt $P(f)$ là nối tầng của hai bộ lọc mỗi bộ lọc có hàm truyền đạt $\sqrt{P(f)}$. Một được dùng ở máy phát và một được dùng ở máy thu. Dẫn đến:

$$p_{SQRC}(t) = 4\beta \frac{\cos\left[(1+\beta)\pi t/T\right] + \sin\left[(1-\beta)\pi t/T\right]\left(4\beta t/T\right)^{-1}}{\pi\sqrt{T}\left[1 - 16\beta^2 t^2/T^2\right]} \tag{5.88}$$

Biểu thức (5.85) được áp dụng để thực hiện trễ và lấy mẫu. Kết quả là:

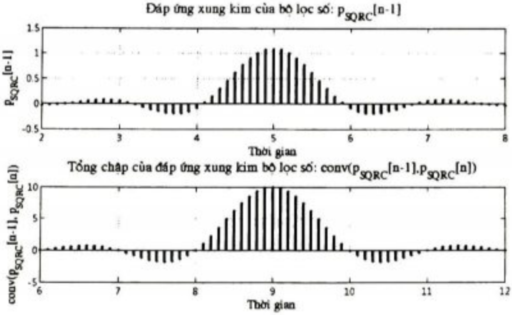
$$p_{SQRC}[n] = 4\beta \frac{\cos\{(1+\beta)\pi[(n/k)-m]\} \; + \; \sin\{(1-\beta)\pi[(n/k)-m]\} \times \{4\beta[(n/k)-m]\}^{-1}}{\pi\sqrt{T}\left\{1 - 16\beta^2\left[(n/k)-m\right]^2\right\}} \tag{5.89}$$

Hình 5.6 (phần trên) minh họa đáp ứng của bộ lọc có đáp ứng xung kim $p_{SQRC}[n]$ khi đặt vào $\delta[n-1]$. Nó là:

$$p_{SQRC}[n]\otimes\delta[n-1]=p_{SQRC}[n-1] \tag{5.90}$$

Tham số bộ lọc là $\beta=0{,}32$, $k=10$ mẫu trên một ký hiệu, T = 1, m = 4 ký hiệu. Lưu ý: *(i)* điểm cắt không xảy ra không chính xác tại bội số nguyên lần của T = 1. Nửa dưới của hình 5.16 minh họa kết quả lấy tích chập của chuỗi mẫu (được cho ở hình 5.16) với $p_{SQRC}[n]$. Về cơ bản nó thể hiện tích chập của $p_{SQRC}[n]$ với chính nó và tương đương với hai bội lọc SQRC trong nối tầng. *(ii)* các điểm không bây giờ rơi vào các vị trí là bội số nguyên lần chu kỳ T = 1 vì kết hợp xếp tầng của hai bộ lọc SQRC là một bộ lọc ISI = 0. Mã chương trình Matlab được cho ở file **NVD5_sqrcsim.m** trong Phụ lục 5A thực hiện tính toán các kết quả trên.

Kết quả nghiên cứu phần này được ứng dụng ở chương 18 khi đó ta mô phỏng hệ thống thông tin vệ tinh.



Hình 5.16: Xung cosin tăng căn bậc 2: ***SQRC***

5.5.3. Thực hiện mô hình mô phỏng bộ lọc FIR

Các mô hình FIR đóng vai trò trung tâm trong mô phỏng hệ thống truyền thông. Trong các phần trước, kỹ thuật thiết kế được sáng tỏ cho trường hợp hàm truyền đạt hay đáp ứng xung kim là đã được biết trước ở dạng giải tích do đó nó được lấy mẫu. Một điểm quan trọng nữa khi sử dụng mô hình mô phỏng FIR là các giai đoạn sau của quá trình thiết kế. Khi này, bộ lọc đã thực sự được thiết kế và xây dựng vì vậy đáp ứng tần số là khả dụng. Các mô hình mô phỏng FIR rất phù hợp để mô phỏng bộ lọc có đáp ứng tần số được cho ở dạng đo đạc.

Ta tóm tắt các bước quan trọng để mô phỏng bộ lọc có đáp ứng tần số tùy ý (đáp ứng tần số này được xác định ở dạng bảng số liệu đo kiểm). Muốn vậy, ta tiến hành ba bước cơ bản là: chọn tham số, lập mô hình mô phỏng và thực hiện mô phỏng bằng Matlab:

Bước 1 (chọn tham số): Chọn hai tham số quan trọng cơ bản là: tần số lấy mẫu và khoảng thời gian của đáp ứng xung kim đã được cắt. Các tham số thực thi bộ lọc FIR phải được

chọn cẩn thận để giảm thiểu mức độ phức tạp tính toán đồng thời vẫn đáp ứng sự phân giải trong miền thời gian và tần số. Ta nên chọn các tham số theo trình tự sau:

- Tần số lấy mẫu f_s: $f_s > (16B \div 32B)$, B là độ rộng băng của bộ lọc. Độ phân giải thời gian (thời gian giữa các mẫu) T_s là $1/f_s$:

- Phân giải tần số, $\Delta_f = \frac{1}{NT_s} = \frac{f_s}{N}$, trong đó $N = \frac{f_s}{\Delta_f}$. Giá trị hay dùng là $\Delta_f = \left(\frac{B}{64} \div \frac{B}{32}\right)$.

- Số lượng mẫu/ký hiệu là số nguyên và là lũy thừa của 2. Giá trị nhỏ nhất là 8.

- Khoảng thời gian của đáp ứng xung kim thường từ 8 đến 6 ký hiệu.

Từ đó, dẫn đến lựa chọn N = 1024 với N là số lượng nhánh bộ lọc hay là số lượng mẫu của đáp ứng xung kim.

Bước 2 (lập mô hình): Sau khi chọn các tham số chủ đạo là f_s và N (do vậy ta có T_s và Δ_f) sau đó thực hiện tiền xử lý số liệu đáp ứng tần số và tiến hành lập mô hình mô phỏng như sau:

- *Trước tiên là tiền xử lý số liệu đáp ứng tần số*: Bước này bao gồm chuyển thông dải thành thông thấp (nếu bộ lọc là bộ lọc thông dải), lấy mẫu lại số liệu đáp ứng tần số và chuyển trễ nhóm thành pha bằng cách tích phân dữ liệu trễ nhóm. Việc chuyển thông dải thành thông thấp bằng cách đánh nhãn lại trục tần số ($f_l = f - f_c$). Dẫn đến tạo ra một tương đương thông thấp không đối xứng liên hợp qua $f = 0$. Khi này, đáp ứng xung kim của bộ lọc tương đương thông thấp có thể là giá trị phức.
- *Tiếp đó là lấy mẫu lại:* Cần phải lấy mẫu lại nếu các kết quả đo phân tích mạng được cho tại các điểm tần số khác với dữ liệu được sử dụng để thực thi FIR. Chẳng hạn, giả sử có 100 mẫu đáp ứng tần số được tập hợp từ máy đo phân tích mạng và việc thực thi FIR dựa trên N = 1024 mẫu. Khi này, đáp ứng tần số phải được nội suy. Nội suy tuyến tính đơn giản thích hợp trong hầu hết các trường hợp.
- *Cuối cùng là chuyển đáp ứng trễ nhóm thành đáp ứng pha:* Một lần nữa, số liệu trễ nhóm được chuyển thành đáp ứng pha bằng cách lấy tích phân trễ nhóm theo tần số.

Bước 3 (thực thi bằng Matlab): Một khi đáp ứng tần số của phiên bản tương đương thông thấp của bộ lọc là khả dụng thì việc thực thi mô hình bộ lọc FIR trên Matlab gồm các bước sau đây:

1. Mở rộng số liệu đáp ứng tần số từ $-f_s/2$ đến $f_s/2$ và có được các giá trị mẫu của hàm truyền đạt $H(f)$, $-f_s/2 < f < f_s/2$ với $f = k\Delta_f$, $k = -N/2$ đến $N/2 - 1$, Δ_f là độ phân giải tần số ($\Delta_f = 1/(NT_s)$).
2. Di chuyển phần phổ âm của *H(f)* từ $N/2 + 1$ tới N sao cho đáp ứng tần số được chứa trong $H(k\Delta_f)$, $k = 1, 2, ..., N$.
3. Lấy biến đổi IFFT để có được đáp ứng xung kim (lấy mẫu trong miền thời gian sẽ là $T_s = 1/(N\Delta_f)$).

4. Cửa sổ hóa hàm đáp ứng xung kim nếu cần thiết (phải chắc chắn rằng đáp ứng xung kim được "quay" chính xác và được trung tâm trong hàm cửa sổ). Chuẩn hóa đáp ứng xung kim đã được cửa sổ hóa để có năng lượng đơn vị.

Sau khi có được đáp ứng xung kim đã được cắt ngắn, mô phỏng bộ lọc trên Matlab bằng hàm *filter* với các tham số a = 1 và b =.h, h là dãy đáp ứng xung kim, nghĩa là đầu ra bộ lọc được tính toán bằng dòng lệnh:

$$\text{Đầu ra} = \text{filter}\left(\text{b, a, đầu vào}\right)$$

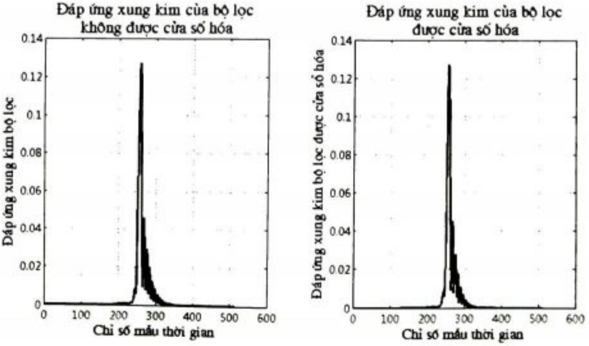
Nếu chuỗi đầu vào quá dài thì ta tính chuỗi đầu ra bằng hàm **fftfilt.m** như sau:

$$\text{Đầu ra} = \text{fftfilt}\left(\text{b, đầu vào, N}\right)$$

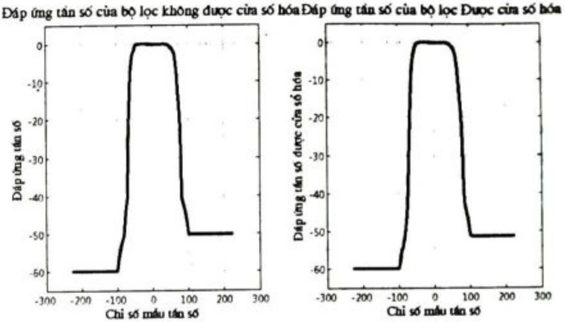
Thực thi bộ lọc theo khối bằng cách dùng phương pháp chồng lấn và cộng với kích thước khối là N như đã được đề cập.

Chương trình Matlab minh họa thực thi bài toán này được cho ở file **NVD5_FIRExample.m** trong Phụ lục 5A, cho thấy làm thế nào mô phỏng một bộ lọc định kênh thông dải 125 MHz được sử dụng trong vệ tinh. Các đặc tính dữ liệu đo được cho ở dạng các đặc tính trễ và đáp ứng biên độ (xem file **NVD5_Filter_Data.m** trong Phụ lục 5A).

Ví dụ 5.11: Chương trình Matlab để thiết kế bộ lọc FIR sử dụng kỹ thuật đã mô tả được cho ở file **NVD5_FIRExample.m** trong Phụ lục 5A. Số liệu đo được dùng để thiết kế cũng được cho ở **NVD5_Filter_Data.m** trong Phụ lục 5A. Kết quả mô phỏng được minh họa ở hình 5.17 và 5.18, với hình 5.17 minh họa đáp ứng xung kim được cửa số hóa và không được cửa sổ hóa. Đáp ứng được cửa sổ hóa (cắt ngắn) và không được cửa số hóa (không bị cắt ngắn) được vẽ độc lập bởi lẽ chúng khá giống nhau và sẽ không thể phân biệt được nếu để trong cùng một hình vẽ. Điểm khác nhau duy nhất giữa hai đáp ứng biên độ này là vùng phẳng trong phần vùng lân cận chỉ số mẫu từ 150 đến 200. Sự suy hao là lớn hơn một ít khi dùng hàm cửa sổ Hamming.



Hình 5.17: So sánh đáp xung (được cửa sổ hóa và không được cửa sổ hóa)



Hình 5.18: So sánh đáp ứng biên độ (được cửa sổ hóa và không được cửa sổ hóa)

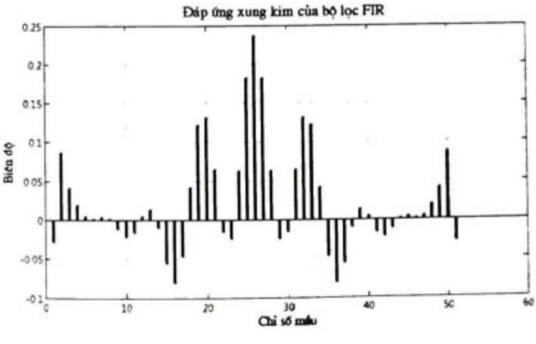
5.5.4. Thiết kế bộ lọc số FIR với sự hỗ trợ của máy tính

Tính phổ cập nhất của kỹ thuật thiết kê trợ giúp máy tính CAD để thiết kế bộ lọc FIR là giải thuật Parks-McClellan, có thể áp dụng cho bộ lọc có đặc tính đáp ứng biên độ lý tưởng hằng số. Chẳng hạn áp dụng thuật toán Parks-McClellan để tổng hợp bộ lọc định kênh có đặc tính đáp ứng biên độ được cho ở hình 5.9. Thuật toán Parks - McCllellan sử dụng đa thức Chebyshev làm phù hợp với đáp ứng biên độ mong muốn và là xấp xỉ hóa đẳng gợn (độ gợn sóng như nhau) với đáp ứng biên độ mong muốn. Nghĩa là, về phương diện tỉ lệ tuyến tính, biên độ độ gọn sóng của dải thông và dải chắn là bằng nhau.

Ví dụ 5.12: Minh họa thiết kế dựa trên máy tính CAD bộ lọc FIR và thuật toán Parks-McClellan, xét đáp ứng biên độ mong muốn được cho ở hình 5.9. Chương trình Matlab để xác định xung kim của bộ lọc được cho bởi file **NVD5_pmc.m** trong Phụ lục 5A.

Lưu ý rằng, các véc-tơ xác định các điểm tần số và đáp ứng biên độ mong muốn tại các điểm tần số của chúng là chung cho cả 2 chương trình tổng hợp bộ lọc FIR và IIR.

Kết quả chạy chương trình mô phỏng là đáp ứng xung kim được minh họa ở hình 5.19. Lưu ý rằng, có 51 thành phần trong đáp ứng xung kim, tương ứng với bậc là 50 và đáp ứng xung kim này là hàm chẵn qua thành phần trung tâm. Thực thi bộ lọc ở dạng đường trễ rẽ nhánh TDL

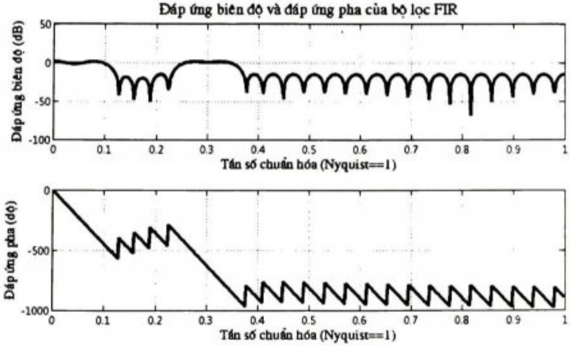


Hình 5.19: Đáp ứng xung kim của bộ lọc FIR

với các giá trị được minh họa trong hình 5.19 là các trọng số nhánh. Thiết kế bộ lọc "hoạt động độc lập" hoặc liên kết với các chương trình mô phỏng sao cho quá trình thiết kế bộ lọc trở thành một phần của quá trình mô phỏng.

Đáp ứng pha và biên độ của bộ lọc số được minh họa ở hình 5.20. Lưu ý rằng, độ gợn dải thông và dải chắn là đặc điểm của quá trình thiết kế Parks - McClellan. Cho thấy, suy hao dải chắn khoảng 20 dB. Suy hao dải chắn có thể tăng bằng cách tăng tăng bậc của bộ lọc. Tăng bậc của bộ lọc cũng giảm độ gợn của dải thông. Yêu cầu về bậc của bộ lọc được xác định bởi độ gợn dải thông cho phép (độ gợn dải thông càng nhỏ thì yêu cầu bậc bộ lọc càng lớn), suy hao dải chắn (suy hao dải chắn càng lớn yêu cầu bậc bộ lọc càng lớn) và độ rộng của băng tần chuyển dịch (băng tần chuyển dịch càng hẹp yêu cầu bậc bộ lọc càng lớn). Thiết kế điển hình sử dụng quá trình Park-McClellan là một quá trình lặp trong đó các chỉ tiêu được điều chỉnh trong giới hạn cho phép với hy vọng sẽ nhận được đáp ứng xung kim có chiều dài thích hợp.

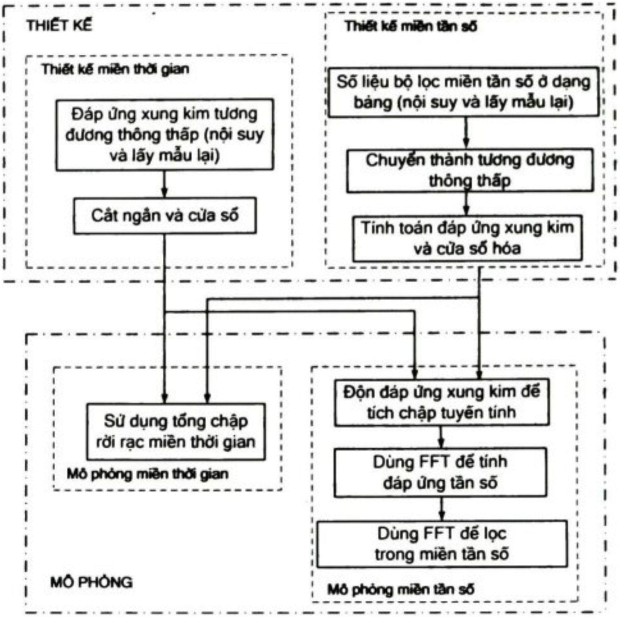
Vì đáp ứng xung kim có dạng hàm chẵn bị trễ, nên đáp ứng pha sẽ là tuyến tính như ở hình 5.20. Nhẩy 180^0 trong đáp ứng pha tương ứng với đổi dấu trong đáp ứng biên độ khi đáp ứng biên độ đi qua 0 ($-\infty$ dB). Độ dốc của đáp ứng pha, trễ nhóm của bộ lọc được xác định bởi trễ cần thiết để tạo ra đáp ứng xung kim của bộ lọc có tính nhân quả.



Hình 5.20: Đáp biên độ và pha của bộ lọc FIR

5.5.5. Bình luận về thiết kế bộ lọc FIR

Như đã đề cập, có thể thiết kế các bộ lọc FIR mà không cần có tương đương tương tự. Điều này càng có ý nghĩa bởi lẽ xu thế các hệ thống truyền thông dựa vào phần mềm (chẳng hạn: vô tuyến được định nghĩa bằng phần mềm hay mềm hóa cấu hình hệ thống...) với việc số hóa bộ lọc. Ở dạng thực thi, các ưu, nhược điểm quan trọng nhất được cho ở bảng 5.2. Hình 5.21 tóm tắt các kỹ thuật mô phỏng và thiết kế bộ lọc FIR. Lưu ý rằng, có thể thiết kế sử dụng số liệu trong miền tần số hoặc miền thời gian. Việc mô phỏng trong miền thời gian hay mô phỏng trong miền tần số tùy vào việc chọn số liệu thiết kế.



Hình 5.21: Thiết kế bộ lọc FIR và các kỹ thuật mô phỏng

Bảng 5.2: Các ưu nhược điểm của các kỹ thuật thực thi bộ lọc FIR

Thực hiện	Ưu điểm	Nhược điểm
Miền thời gian	Thực thi đơn giản	Mất nhiều thời gian với chuỗi đáp ứng xung kim dài.
Miền tần số	Xử lý nhanh Dễ dàng thiết kế bằng cách sử dụng dữ liệu đáp ứng tần số.	Gây trễ nhân tạo bằng với chiều dài khối FFT. Không thể sử dụng trong hệ thống có hồi tiếp.
Phương pháp CAD	Dẫn đến bộ lọc pha tuyến tính	Đáp ứng xung kim dài.

Chương 6
VÒNG KHÓA PHA
VÀ PHƯƠNG PHÁP GIẢI PHƯƠNG TRÌNH VI PHÂN

6.1. Mở đầu

Dù vẫn còn sớm trong nghiên cứu mô phỏng nhưng ta có công cụ cần thiết để: sử dụng hiệu quả mô phỏng, giám sát việc thiết kế và các đặc tính hoạt động của một số hệ thống quan trọng. Mặc dù tồn tại nhiều hệ thống lựa chọn cho vấn đề này, nhưng trọng tâm của chương là vòng khoá pha PLL, có một vài lý do cho sự lựa chọn này. *Trước hết*, PLL là một trong những *khối cơ bản* nền tảng trong nhiều phân hệ của hệ thống truyền thông hiện đại. PLL được dùng rộng rãi trong: bộ tổng hợp tần số, bộ chia tần số, bộ nhân tần số, để đồng bộ sóng mang và ký hiệu ở các máy thu nhất quán. Do tính *phi tuyến* của PLL nên rất khó phân tích và thiết kế theo các phương pháp truyền thống (không dựa vào mô phỏng) vì vậy mô phỏng là giải pháp được sử dụng rộng rãi để thiết kế và phân tích. Cần lưu ý rằng, kiến thức cơ bản về phương pháp tích phân số cung cấp những khái niệm cần thiết để triển khai mô phỏng hiệu quả cho các PLL và cho nhiều hệ thống quan tâm khác. *Một lý do khác* để sử dụng PLL trong mô phỏng là PLL chứa đường hồi tiếp. Ta sẽ thấy rằng các hệ thống chứa đường hồi tiếp phải được mô phỏng cẩn thận, đặc biệt là tần số lấy mẫu nếu muốn giảm thiểu lỗi mô phỏng.

Như đã đề cập ở chương 4, việc phân tích dựa trên các mô hình hệ thống thông thấp là tương đương với các mô hình hệ thống băng thông tương ứng, tất nhiên các mô hình thông thấp phải được triển khai chính xác. Ngoài ra, thời gian thực hiện mô phỏng ở các mô hình mô phỏng được rút ra từ những mô hình phân tích thông thấp nhanh hơn nhiều. Cuối cùng, tất cả các mô phỏng được trình bày ở chương này là những mô phỏng *tất định* như được đề cập ở chương 1.

Mặc dù trọng tâm của chương 6 là minh hoạ một số kỹ thuật mô phỏng cơ bản và chỉ sử dụng PLL phục vụ nghiên cứu hoạt động truyền thông nhưng lý thuyết được trình bày cũng đủ để hiểu và "kiểm tra tính đúng đắn" các kết quả mô phỏng. Chương này được bắt đầu bằng việc mô tả cơ bản về PLL. Sử dụng vòng bậc nhất làm ví dụ minh họa, ta sẽ chỉ ra rằng làm thế nào vòng đạt được khóa pha, sau đó mở rộng nghiên cứu cho vòng bậc hai. Dùng mô phỏng để minh hoạ hiện tượng trượt tuần hoàn và xác định thời gian cần thiết để đạt được khoá pha. Mô phỏng được thực hiện bằng cách triển khai mô hình mô phỏng rời rạc theo thời gian cho PLL.

Phần cuối của chương chỉ ra, không những có được các công cụ để nghiên cứu PLL mà còn có được công cụ mô phỏng cần thiết để nghiên cứu hệ thống được định nghĩa ở dạng phương trình vi phân. Phương trình vi phân, hệ thống tương ứng có thể phi tuyến thay đổi theo thời gian (vừa phi tuyến vừa thay đổi theo thời gian). Vì vậy kết thúc chương này, ta sẽ có các công cụ cần thiết để mô phỏng nhiều hệ thống phức tạp khác nhau.

Lưu ý rằng: Nghiên cứu chi tiết về PLL và ứng dụng trong việc đồng bộ sóng mang và định thời ký hiệu được cho ở Phụ lục 6B

6.2. Khái niệm vòng khoá pha cơ bản

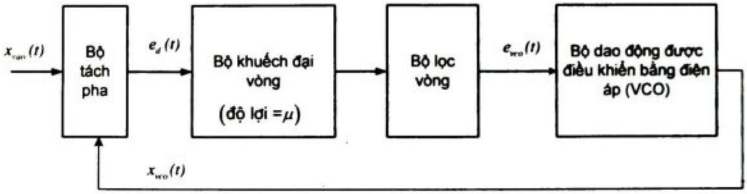
Mô hình cơ bản của PLL được minh hoạ ở hình 6.1, tín hiệu vào được giả định là:

$$x_{vào}(t)= A_c \cos\left[2\pi f_c t + \phi(t)\right] \quad (6.1)$$

Và tín hiệu tại đầu ra của bộ dao động được điều khiển bằng điện áp (VCO) được giả định có dạng:

$$x_{vco}(t)=- A_v \sin\left[2\pi f_c t + \theta(t)\right] \quad (6.2)$$

Vai trò cơ bản của PLL là đồng bộ pha của VCO với pha của tín hiệu vào sao cho lỗi pha $\phi(t)-\theta(t)$ là nhỏ nhất có thể. Ta sẽ làm rõ, cách thực hiện vấn đề này trong phần sau.



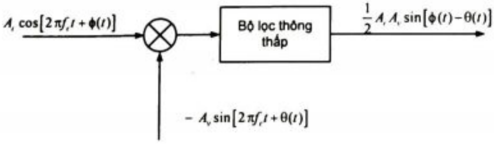
Hình 6.1: Sơ đồ khối vòng khóa pha PLL

PLL thường hoạt động ở một trong hai chế độ, *bắt* và *bám*. Trong chế độ bắt, PLL thử để đồng bộ tần số và pha của tín hiệu ra của VCO với tín hiệu vào. Khi này, các lỗi pha có thể rất lớn, khi này rõ ràng hoạt động của PLL là phi tuyến và cần phải phân tích mô hình phi tuyến. Phân tích các mô hình *phi tuyến* là rất khó và thường cần đến mô phỏng. Tuy nhiên trong chế độ *bám*, lỗi pha thường nhỏ trong những khoảng thời gian dài, và phân tích sử dụng mô hình tuyến tính đơn giản thường cho các kết quả thoả đáng mà không cần đến mô phỏng. Ta sẽ làm rõ, các tham số vòng tiêu chuẩn được định nghĩa theo mô hình *tuyến tính,* vì vậy cần phải quan tâm cả mô hình tuyến tính và phi tuyến. *Trọng tâm của các mô phỏng được triển khai trong chương này sẽ là tính cách phi tuyến của hệ thống.*

6.2.1. Các mô hình vòng khóa pha

Bước 1: Mô hình hóa bộ tách pha

Các đặc điểm của bộ tách pha phần lớn xác định những đặc điểm hoạt động của PLL. Tồn tại nhiều loại bộ tách pha, việc chọn mô hình bộ tách pha là phụ thuộc vào ứng dụng. Mô hình bộ tách pha thông dụng nhất là bộ tách pha hình **sin** (là bộ tách pha mà đầu ra của nó tỷ lệ với **sin** của lỗi pha). Bộ tách pha hình sin gồm một bộ nhân và một bộ lọc thông thấp như được minh họa ở hình 6.2. Chức năng duy nhất của bộ lọc thông thấp là loại bỏ hài thứ cấp của tần số sóng mang ở đầu ra bộ nhân. Ta sẽ thấy rằng, trong các ứng dụng thực tế thì bộ lọc thông thấp là không cần thiết.



Hình 6.2: Mô hình bộ tách pha hình ***sin***

Với mô hình bộ tách pha hình 6.2, đầu ra bộ tách pha là:

$$e_d(t) = \frac{1}{2} A_c A_v \sin\left[\phi(t) - \theta(t)\right] \tag{6.3}$$

Trong đó $\phi(t) - \theta(t)$ là lỗi pha. Phần sau sẽ ký hiệu lỗi pha là $\psi(t)$, nhưng ở đây tốt hơn là biểu diễn theo pha đầu vào $\phi(t)$ và pha đầu ra bộ VCO $\theta(t)$. Vì ta muốn pha VCO là một ***bộ ước tính pha đầu vào*** vì vậy hoạt động đúng của PLL là phải điều khiển lỗi pha về 0. Lỗi pha ở trạng thái ổn định có thể bằng 0 hoặc khác 0 phụ thuộc vào các đặc điểm tín hiệu vào và bộ lọc vòng.

Lưu ý rằng đầu vào PLL và đầu ra VCO là vuông pha nhau khi $\phi(t) = \theta(t)$. Điều này cần thiết nếu đầu ra bộ tách pha là *hàm lẻ* của lỗi pha. Dễ dàng thấy rằng nếu các hàm **cos** được sử dụng trong cả (6.1) và (6.2) với đối số không thay đổi thì đầu ra bộ tách pha sẽ tỷ lệ nghịch với $\cos[\phi(t) - \theta(t)]$ (là *hàm chẵn* của lỗi pha). Tất nhiên điều này tạo ra tình huống không mong muốn là không thể phân biệt được các lỗi pha âm với các lỗi pha dương.

Sau khi nhân với độ lợi bộ khuếch đại vòng μ, đầu ra bộ tách pha $e_d(t)$ được lọc bởi bộ lọc vòng có hàm truyền đạt $F(s)$ và đáp ứng xung kim đơn vị $f(t)$. Vì vậy đầu vào VCO được cho bởi:

$$e_{vco}(t) = \int_{-\infty}^{\infty} \mu . e_d(\lambda) . f(t-\lambda) . d\lambda \tag{6.4}$$

Chỉ là *tích chập* của đáp ứng xung kim bộ lọc vòng với đầu vào bộ lọc vòng. Với bộ tách pha hình **sin** thì:

$$e_{vco}(t) = \int_{-\infty}^{\infty} \frac{1}{2} \mu . A_c . A_v . \sin\left[\phi(\lambda) - \theta(\lambda)\right] . f(t-\lambda) d\lambda \tag{6.5}$$

Bước 2: Thiết lập quan hệ độ lệch pha VCO $\theta(t)$ ***với đầu vào VCO***

Theo định nghĩa, độ lệch tần của đầu ra VCO là tỷ lệ nghịch với tín hiệu đầu vào VCO sao cho:

$$\frac{d\theta}{dt} = 2\pi . K_d . e_{vco}(t) \tag{6.6}$$

Trong đó K_d là hằng số VCO và có đơn vị là Hz/v. Giải phương trình để tìm $\theta(t)$ ta được:

$$\theta(t) = 2\pi K_d \int_{-\infty}^{t} e_{vco}(\lambda) d\lambda \tag{6.7}$$

Thay $e_{vco}(t)$ từ (6.5) vào (6.7) ta được:

$$\theta(t) = G.\int_{-\infty}^{t} \int_{-\infty}^{\infty} \sin\left[\phi(\lambda) - \theta(\lambda)\right].f(\tau\text{-}\lambda) d\lambda d\tau \tag{6.8}$$

Trong đó *độ lợi vòng* G được định nghĩa là:

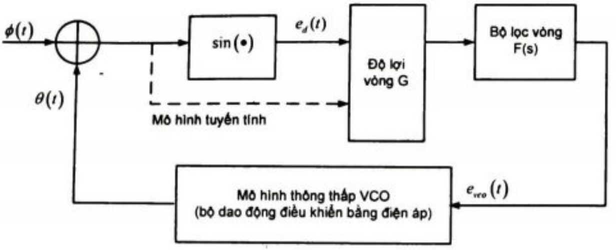
$$G = \pi.K_d.\mu.A_c.A_v \tag{6.9}$$

Phương trình tích phân phi tuyến (6.8) thiết lập quan hệ giữa độ lệch pha của đầu vào $\phi(t)$ với độ lệch pha $\theta(t)$ của VCO. Cần lưu ý rằng, đáp ứng xung kim của bộ lọc vòng $f(t)$ vẫn là bất kỳ.

Khi triển khai mô phỏng hệ thống, nên cẩn thận kết hợp các tham số như ta đã làm trong (6.9). Kết hợp các số hạng là một bước hợp lý nếu mục đích của mô phỏng là xác định các đặc tính vào/ra của hệ thống hoặc các đặc tính của toàn bộ hệ thống, như thời gian cần thiết để PLL đạt được khoá pha. Mặt khác, nếu mục đích của mô phỏng là để kiểm tra sự xuất hiện dạng sóng tại đầu vào/đầu ra các khối chức năng trong hệ thống, thì các tham số định nghĩa cho G không được nhóm lại với nhau. Nếu chúng được nhóm lại với nhau, thì dạng sóng có mặt tại các điểm khác nhau trong hệ thống sẽ không thể lấy tỉ lệ chính xác được hoặc thậm chí không thể nhận dạng.

6.2.2. Mô hình pha phi tuyến

Từ biểu thức (6.8) thấy rõ quan hệ giữa $\theta(t)$ và $\phi(t)$ là không phụ thuộc vào tần số sóng mang fc vì vậy tần số sóng mang không được xét trong mô hình mô phỏng. Theo đó, ta tìm kiếm một mô hình thiết lập quan hệ giữa $\theta(t)$ và $\phi(t)$ mà không cần phải xét tần số sóng mang. Mô hình này được cho ở hình 6.3 là mô hình pha phi tuyến của PLL. Đây là ***mô hình pha phi tuyến*** vì hai lý do: *(i)* là mô hình *phi tuyến* (vì tính phi tuyến của hình **sin**); *(ii)* là mô hình *pha* (vì nó thiết lập quan hệ giữa *độ lệch pha* đầu vào và *độ lệch pha* VCO chứ không phải là thiết lập quan hệ giữa *tín hiệu* đầu ra VCO và đầu vào vòng thực tế như được biểu diễn bởi (6.1) và (6.2)). Cần lưu ý rằng, đưa vào mô hình được minh hoạ trong hình 6.1 là tín hiệu thông băng thực tế, trong khi đó khi đưa vào mô hình pha phi tuyến được cho ở hình 6.3 là độ lệch pha của tín hiệu băng thông đầu vào. Nếu độ lệch pha của tín hiệu đầu vào $\phi(t)$, tần số sóng mang và biên độ tín hiệu được biết, thì (6.1) được xác định hoàn toàn. Tương tự, nếu độ lệch pha $\theta(t)$ của VCO, tần số sóng mang và biên độ được biết trước, thì đầu ra VCO được biểu diễn bởi (6.2) cũng được xác định hoàn toàn. Vì vậy, mô hình pha phi tuyến biểu diễn quan hệ giữa pha đầu vào và pha đầu ra VCO là đại lượng quan trọng. Trong mô phỏng, mô hình pha phi tuyến phải chịu số bị chia bổ sung. Vì độ lệch pha đầu ra VCO và đầu vào vòng là các tín hiệu thông thấp, nên chúng được lấy mẫu tại tốc độ lấy mẫu thấp hơn nhiều so các tín hiệu thông băng được biểu diễn bởi (6.1) và (6.2).



Hình 6.3: Mô hình pha PLL phi tuyến
(mô hình tuyến tính được thể hiện bằng đường nét đứt)

Tại đây, dùng bộ lọc thông thấp để khử hài thứ cấp (do bộ nhân trong mô hình tách pha gây ra). Bộ lọc này chỉ là một phần của mô hình khái niệm, như đã được đề cập, nó không cần xuất hiện trong thiết bị vật lý. Dễ dàng thấy rằng, bộ lọc này có thể được loại bỏ. Phương trình (6.7) cho thấy mô hình VCO chỉ là bộ tích phân. Vì bộ tích phân là bộ lọc thông thấp có độ lợi vô hạn tại $f = 0$ và độ lợi đơn vị tại $f = 1/(2\pi)$ Hz, nên VCO sẽ ngăn chặn hài thứ cấp của tần số sóng mang truyền lan quanh vòng và xuất hiện tại đầu ra VCO.

Ví dụ 6.1: Xét kỹ thuật tổng quát để mô hình hóa bộ tách pha. Mặc dù bộ tách pha hình **sin** được giả định trong suốt chương này, nhưng nó dễ dàng để mô hình hóa bộ tách pha có đặc tính bất kỳ. Kỹ thuật tổng quát để biểu diễn *hàm* liên hệ đầu ra bộ tách pha với đầu vào bộ tách pha là chuỗi Fouriers:

$$e_d(t) = \sum_{k=1}^{N} c_k \sin[k\psi(t)] \tag{6.10}$$

Trong đó c_k biểu diễn các hệ số Fourier của đặc tính bộ tách pha, $\psi(t) = \phi(t) - \theta(t)$ là lỗi pha. Có thể mô hình hóa đặc tính bộ tách pha cho trước với độ chính xác cần thiết bất kỳ bằng cách thay đổi giá trị của N trong khai triển chuỗi. Cần lưu ý đến các thành phần làm cho $e_d(t)$ là *hàm lẻ* của lỗi pha. Ví dụ: bộ tách pha hình **sin** được định nghĩa trong hình 6.2 được biểu diễn bởi:

$$c_k = \begin{cases} \dfrac{1}{2} A_c A_v, & k = 1 \\ & , N \geq 1 \\ 0 & , \textit{nếu khác} \end{cases} \tag{6.11}$$

Mã chương trình Matlab cho bộ tách pha được mô tả bởi (6.10) được viết dễ dàng ở dạng véc tơ. Véc-tơ B được định nghĩa là:

$$\mathbf{B} = [1 \quad 2 \quad 3 \quad \ldots \quad N]$$

Và véc tơ hệ số Fourier là:

$$\mathbf{C} = [c_1 \quad c_2 \quad c_3 \quad \ldots \quad c_N]$$

Giả sử p_{din} và p_{dout} là đầu vào/ra của mô hình bộ tách pha tương ứng, thì đầu ra bộ tách pha có thể được viết là:

$$p_{dout} = \left(\sin\left(B * p_{din}\right)\right) * C'$$

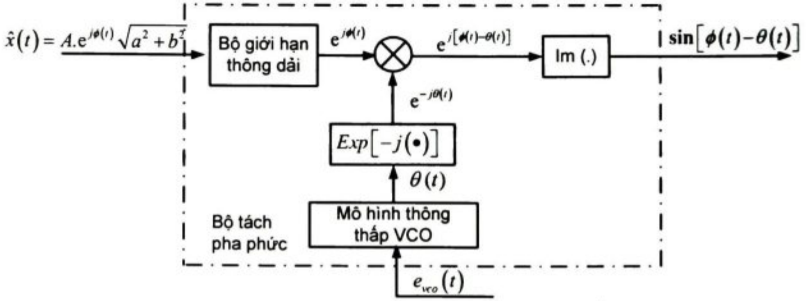
Trong đó C' là chuyển vị của C. Vì B và C hoàn toàn được xác định bởi mô hình bộ tách pha, nên chúng là cố định và ta nên đặt chúng ở ngoài vòng mô phỏng.

Cũng sẽ thấy rõ rằng, phương pháp tổng quát nhất để mô phỏng hệ thống là sử dụng kỹ thuật Monte Carlo, nó cần có thời gian mô phỏng lâu. Vì thời gian mô phỏng theo kỹ thuật Monte Carlo lâu nên cần có các thuật toán hiệu quả nhất có thể. Vì vậy công thức véc tơ được dùng tính toán p_{dout} nhằm tránh hoạt động vòng thường được sử dụng để ước lượng các tổng.

6.2.3. Mô hình phi tuyến với đầu vào phức

Các vòng khoá pha thường được mô hình hóa sao cho đầu vào vòng là độ lệch pha $\phi(t)$. Khi PLL được dùng trong hệ thống, ta thường mong muốn triển khai mô hình trong đó đầu vào vòng là tín hiệu thông thấp đường bao phức $Ae^{j\phi(t)}$ biểu diễn cho (6.1). Một mô hình như vậy được minh hoạ ở hình 6.4 cho bộ tách pha hình **sin**. Các bộ tách pha có các đặc điểm khác dễ dàng được rút ra.

Như được minh hoạ trong hình 6.4, tín hiệu đầu vào ở dạng đường bao phức, trước hết nó được đi qua bộ giới hạn thông dải đã được đề cập trong ví dụ 4.10 chương 4 (ở đây ta giả sử rằng tham số B trong ví dụ 4.10 được đặt bằng 1). Hoạt động của mô hình bộ tách pha phức được sáng tỏ từ các biểu thức trong hình 6.4. Lưu ý rằng, các đầu vào bộ nhân được biểu diễn ở dạng số mũ phức và đầu ra bộ lấy phần ảnh của đầu ra bộ nhân cho ta $\sin\left[\phi(t)-\theta(t)\right]$.



Hình 6.4: Mô hình bộ tách pha thông thấp phức

6.2.4. Mô hình tuyến tính và hàm truyền đạt vòng

Nếu lỗi pha là nhỏ sao cho có thể lấy xấp xỉ tuyến tính sau:

$$\sin\left[\phi(t)-\theta(t)\right] \approx \phi(t)-\theta(t) \tag{6.12}$$

Thì phương trình vòng (6.8) trở thành:

$$\theta(t) = G\int_{-\infty}^{t}\int_{-\infty}^{\infty}[\phi(\lambda) - \theta(\lambda)].f(\tau-\lambda)\,d\lambda\,d\tau \tag{6.13}$$

Kết quả này dẫn đến ***mô hình pha tuyến tính*** của PLL, nó giống với mô hình pha phi tuyến ngoại trừ tính phi tuyến hình **sin** được loại bỏ (đường nét đứt trong hình 6.3).

Lấy biến đổi Laplace (6.13), thừa nhận rằng phép tích phân trong miền thời gian tương đương với phép chia cho s trong miền tần số và phép chập vòng trong miền thời gian tương đương với phép nhân trong miền tần số, theo đó ta được:

$$\Theta(s) = G.[\Phi(s) - \Theta(s)].\frac{F(s)}{s} \tag{6.14}$$

Vì vậy, hàm truyền đạt H(s) liên hệ giữa pha VCO với pha đầu vào được cho bởi:

$$H(s) = \frac{\Theta(s)}{\Phi(s)} = \frac{G.F(s)}{s + G.F(s)} \tag{6.15}$$

Cần phải lưu ý rằng, hàm truyền đạt (6.15) là dựa trên giả định tuyến tính, ở dạng tổng quát thì không còn đúng nữa. *Nói một cách chính xác, không tồn tại các hàm truyền đạt cho các hệ thống phi tuyến.*

Mặc dù mô phỏng thường được dùng để *nghiên cứu tính cách phi tuyến* của PLL, nhưng các mô hình tuyến tính vẫn hữu hiệu trong một vài mục đích bởi lẽ: *Thứ nhất*, các tham số vòng hầu như luôn được định nghĩa theo mô hình tuyến tính. *Thứ hai*, dễ dàng phân tích giải tích dựa trên mô hình tuyến tính và thường giúp kiểm tra tính đúng đắn các kết quả mô phỏng. *Thứ ba*, thường dùng các mô hình tuyến tính để nghiên cứu tính cách *bám* khi đó tín hiệu đang được bám đang thay đổi tại tốc độ thấp hơn độ rộng băng thông của vòng.

6.3. Vòng bậc một và bậc hai

Vì tính ***bắt*** và ***bám*** của PLL phụ thuộc nhiều vào *bậc* của vòng, nên ta xét hai lựa chọn cho hàm truyền đạt bộ lọc vòng. Bậc thực hiện PLL là bằng số điểm cực trong hàm truyền đạt *H(s)* như được cho trong biểu thức (6.15). Theo đó, bậc thực hiện PLL lớn hơn số điểm cực của *F(s)* là 1. Tại đây ta xét ngắn gọn PLL bậc nhất (không có bộ lọc vòng) và PLL bậc hai (có bộ lọc vòng bậc nhất). Dù PLL bậc nhất không có các đặc tính hoạt động mong muốn, nhưng ta sẽ sử dụng PLL bậc nhất để minh hoạ cơ chế vòng nào đạt được khoá pha.

6.3.1. PLL bậc một

Đối với PLL bậc một $F(s) = 1$ sao cho:

$$f(t) = \delta(t) \tag{6.16}$$

Thay vào (6.8) ta được:

$$\theta(t) = G\int_{-\infty}^{t}\int_{-\infty}^{\infty}\sin\left[\phi(\lambda) - \theta(\lambda)\right].\delta(\tau-\lambda)\,d\lambda\,d\tau \tag{6.17}$$

Lấy tích phân theo biến λ dùng tính chất dịch của hàm xung kim (delta) ta được:

$$\theta(t) = G\int_{-\infty}^{t}\sin\left[\phi(\tau) - \theta(\tau)\right]d\tau \tag{6.18}$$

Lấy vi phân theo t nhận được phương trình vi phân:

$$\frac{d\theta}{dt} = G\sin\left[\phi(t) - \theta(t)\right] \tag{6.19}$$

Viết (6.19) theo lỗi pha $\psi(t) = \phi(t) - \theta(t)$ cho ta phương trình vi phân liên hệ giữa lỗi pha và pha đầu vào. Phương trình là:

$$\frac{d\phi}{dt} - \frac{d\psi}{dt} = G.\sin\psi(t) \tag{6.20}$$

Tại đây ta xác định lỗi pha đối với bước nhảy tần số đầu vào.

Để nghiên cứu đáp ứng của PLL bậc nhất đối với bước nhảy tần số f_Δ Hz tại thời điểm t_0 ta đặt:

$$\frac{d\phi}{dt} = 2\pi f_\Delta u(t - t_0) \tag{6.21}$$

Sao cho (6.20) với $t > t_0$ trở thành:

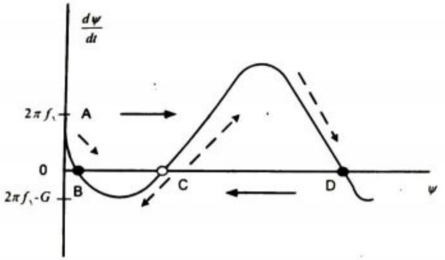
$$\frac{d\psi}{dt} = 2\pi f_\Delta - G.\sin\psi(t) \tag{6.22}$$

Cho ta quan hệ giữa lỗi tần số và lỗi pha khi $t > t_0$.

Phương trình (6.22) được minh hoạ trong hình 6.5 và được gọi là *phương trình mặt phẳng pha* (gọi đơn giản là mặt phẳng pha) mô tả tính cách *động* của hệ thống. Mặt phẳng pha có một số tính chất quan trọng, hiểu biết chúng sẽ hiểu sâu sắc cách cũng như điều kiện để vòng đạt được khoá. Lưu ý rằng, tại mọi thời điểm quan hệ giữa lỗi pha và lỗi tần số phải thoả mãn (6.22). Những điểm phụ thuộc thời gian này được coi là những điểm hoạt động. Trong mặt phẳng pha nửa trên, điểm đang hoạt động di chuyển từ trái sang phải và trong mặt phẳng pha nửa dưới, điểm đang hoạt động di chuyển từ phải sang trái. Điều này là dễ dàng thấy được. Trước hết ta đặt:

$$\frac{d\psi}{dt} \approx \frac{\Delta\psi}{\Delta t} \tag{6.23}$$

Trong đó $\Delta\psi$ và Δt biểu thị cho gia tăng nhỏ về lỗi pha và thời gian. Rõ ràng $\Delta t > 0$ với mọi t vì thời gian luôn tăng. Theo đó, $d\psi > 0$ trong mặt phẳng pha nửa trên và $d\psi < 0$ trong mặt phẳng pha nửa dưới. Vì vậy, lỗi pha tăng (di chuyển từ trái sang phải) trong mặt phẳng pha nửa trên và lỗi pha giảm (di chuyển từ phải sang trái) trong mặt phẳng pha nửa dưới. Điều này được minh hoạ bởi mũi tên liền trong hình 6.5. Điểm hoạt động ổn định (dừng) có thể chỉ nằm trên biên giới giữa hai mặt phẳng pha này. Tất nhiên, điều này chứng tỏ rằng lỗi pha là hằng số (tương đương lỗi tần số bằng 0). Điểm hoạt động là ổn định, nếu sau dao động nhỏ điểm hoạt động trở về vị trí ban đầu của nó. Nếu dao động nhỏ dẫn đến điểm hoạt động di chuyển tới vị trí mới thì điểm hoạt động ban đầu được gọi là không ổn định. Vì vậy, các điểm B và D trong hình 6.5 là các điểm hoạt động ổn định và điểm C là điểm hoạt động không ổn định. Sự di chuyển của các điểm hoạt động được chỉ ra bởi các đường nét đứt trong hình 6.5.



Hình 6.5: Mặt phẳng pha bậc một

Từ (6.22) cho thấy, nếu $2\pi f_\Delta < G$ thì điểm hoạt động ở trạng thái ổn định là điểm ổn định B với giả thiết điểm hoạt động khởi đầu do bước nhảy tần số là A. Tại điểm này lỗi tần số bằng 0 và lỗi pha ở trạng thái ổn định là nghiệm của phương trình (6.22) với $d\psi / dt = 0$. Điều này cho ta lỗi pha ở trạng thái ổn định là:

$$\psi_{ss} = \sin^{-1}\left(\frac{2\pi f_\Delta}{G}\right) \tag{6.24}$$

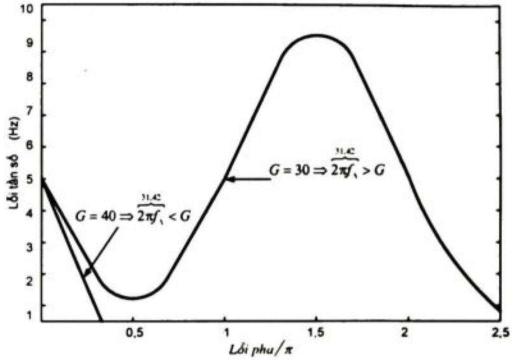
Như sự quan sát cuối cùng về mặt phẳng pha, lưu ý rằng nếu $2\pi f_\Delta > G$ thì không có lời giải cho (6.22) để lỗi tần số bằng 0 $d\psi/dt$ và điểm hoạt động sẽ di chuyển sang bên phải $\forall t$ khi $f_\Delta > 0$ và sẽ di chuyển sang trái $\forall t$ khi $f_\Delta < 0$. Vì vậy độ lợi vòng G trở thành *khoảng khoá* cho vòng bậc nhất. Lưu ý từ (6.15) rằng G cũng là độ rộng băng thông vòng (rad/s) đối với PLL bậc nhất.

Ví dụ 6.2: Như một ví dụ đơn giản, giả sử: *(i)* $f_\Delta = 5$ sao cho $2\pi f_\Delta = 31{,}42$; *(ii)* độ lợi vòng lấy hai giá trị là G = 30 và G = 40.

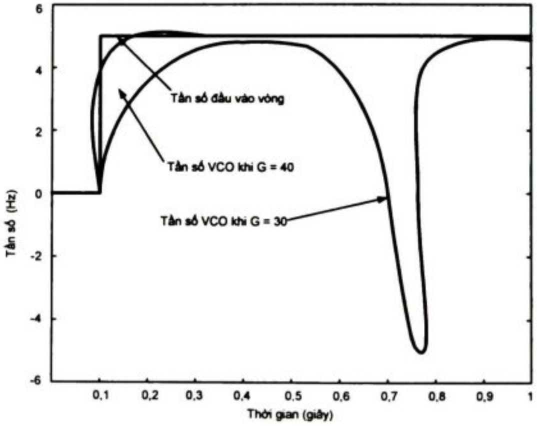
Kết quả, các mặt phẳng pha được cho ở hình 6.6, nó cho thấy: *(i)* khi $f_\Delta = 5$; $G = 40 \Rightarrow 2\pi f_\Delta < G$ thì lỗi tần số giảm đều đều tới 0; *(ii)* Khi $f_\Delta = 5$; $G = 30 \Rightarrow 2\pi f_\Delta > G$ thì (6.22) không có lời giải để lỗi tần số bằng 0. Đối với trường hợp này, không đạt được khoá pha và hệ thống sẽ mãi mãi dao động.

Kết quả, mặt phẳng pha được tạo ra rõ hơn ở hình 6.7, nó cho thấy độ lệch tần số đầu vào $d\phi/dt$ và độ lệch tần số VCO $d\theta/dt$ đối với G = 30 và G = 40. Kết quả, độ lệch tần số đầu vào, độ lệch tần số VCO được cho ở hình 6.7 khi G = 30 và G = 40. Lưu ý rằng, với G = 40 thì vòng đạt được khoá pha, trong khi đó G = 30 thì không đạt được khoá pha và độ lệch pha VCO dao động mãi mãi.

Ta đã thấy rằng, cả *khoảng khoá* của PLL bậc nhất và *độ rộng băng thông vòng* đều được xác định bởi tham số G. Với hầu hết các ứng dụng, đều mong muốn khoảng khoá lớn và băng thông vòng nhỏ. Điều này không thể thực hiện được bằng vòng bậc nhất, vì vậy PLL bậc nhất thường không thực tế. Vòng bậc hai là hệ thống thực tế cho nhiều ứng dụng và sẽ nhận được thu hút nghiên cứu mô phỏng.



Hình 6.6: Mặt phẳng pha đối với PLL bậc một (G = 30 và G = 40)



Hình 6.7: Tần số vào và tần số VCO khi G = 30 và G = 40

6.3.2. PLL bậc hai

PLL bậc nhất có *khoảng khoá bị giới hạn*. Ngoài ra, vì vòng chỉ có một tham số duy nhất, nên khả năng điều chỉnh các tham số vòng để thoả mãn một tập các đặc tính kỹ thuật cho trước bị hạn chế nhiều. Các đặc tính hoạt động được cải thiện và những khả năng thiết kế đạt được bằng cách thay đổi bộ lọc vòng sao cho nhận được PLL bậc hai.

Dạng tổng quát của bộ lọc vòng đối với PLL bậc hai là:

$$F(s) = \frac{s+a}{s+\lambda a} \tag{6.25}$$

PLL bậc hai hoàn hảo được định nghĩa bởi $\lambda = 0$, nó thể hiện cho bộ lọc vòng chứa tích phân lý tưởng (điểm cực tại s = 0). Trong các ứng dụng điển hình thì $\lambda << 1$. Thay hàm truyền đạt bộ lọc vòng vào (6.15), ta có:

$$H(s) = \frac{G.(s+a)}{s^2 + (G + \lambda a).s + G.a} \tag{6.26}$$

đối với hàm truyền đạt mô hình tuyến tính vòng.

Các hệ thống bậc hai tuyến tính thường được tham số hóa theo hệ số tắt dần ς và tần số tự nhiên ω_n của hệ thống. Mẫu số của hàm truyền đạt được gọi là đa thức đặc trưng, với các hệ thống bậc hai được biểu diễn ở dạng chuẩn là:

$$s^2 + 2\varsigma\omega_n s + \omega_n^2$$

Cân bằng đa thức đặc trưng trong (6.26) theo dạng chuẩn ta được:

$$s^2 + (G + \lambda a)s + Ga = s^2 + 2\varsigma\,\omega_n s + \omega_n^2 \tag{6.27}$$

Trong các ứng dụng điển hình, PLL được thiết kế đối với hệ số tắt dần và tần số tự nhiên *cho trước* bằng cách định rõ ς và ω_n. Các tham số vật lý cần thiết (trong trường hợp này là G và a) được xác định sao cho đạt được các giá trị thiết kế ς và ω_n. Cân bằng các thành phần có cùng số mũ của s trong (6.27) ta được:

$$2\varsigma\,\omega_n = G + \lambda a \tag{6.28}$$

$$\omega_n^2 = Ga \tag{6.29}$$

Giả sử λ là hằng số đã biết, ta có thể giải phương trình (6.29) để tìm a và thay kết quả vào (6.28), ta có:

$$G^2 - 2\varsigma\,\omega_n G + \lambda\,\omega_n^2 = 0 \tag{6.30}$$

Tìm được G:

$$G = \varsigma\,\omega_n + \omega_n\sqrt{\varsigma^2 - \lambda} \tag{6.31}$$

Kết quả này cùng với (6.29) cho ta:

$$a = \frac{\omega_n}{\varsigma + \sqrt{\varsigma^2 - \lambda}} \tag{6.32}$$

Lưu ý rằng vì a là tham số giá trị thực nên λ phải nhỏ hơn ς^2. Các giá trị điển hình của ς^2 nằm trong lân cận $\frac{1}{2}$ ($\zeta = 1/\sqrt{2}$ là lựa chọn chung) và $\lambda << 1$. Lưu ý rằng đối với PLL bậc hai hoàn hảo thì $G = 2\varsigma\omega_n$, $a = \dfrac{\omega_n}{2\varsigma}$.

6.4. Mô phỏng vòng khóa pha PLL

Tại đây, ta đang ở giai đoạn mô phỏng PLL bậc hai và sử dụng kiến thức rất cơ bản về lý thuyết PLL để kiểm tra sự đúng đắn của các kết quả. Tuy nhiên, *trước hết* ta xét kiến trúc đơn

giản cho chương trình mô phỏng toàn diện. *Sau đó* triển khai mô hình mô phỏng và thực hiện mô phỏng. *Cuối cùng*, đề cập ngắn gọn các nguồn gây lỗi trong mô phỏng.

6.4.1. Kiến trúc mô phỏng

Thường là hữu ích khi chia phần mềm mô phỏng thành một vài chương trình riêng biệt như ở hình 6.8 gồm 3 phần tử chương trình mô phỏng, được định rõ là bộ tiền xử lý, phương tiện mô phỏng và bộ hậu xử lý. Ba chương trình này thực hiện 3 chức năng riêng biệt và có thể được triển khai trong 3 ngôn ngữ khác nhau. Phân chia mô phỏng theo cách này cho phép dễ dàng triển khai mã chương trình mô phỏng, dễ hiểu và dễ quản lý. Nó cũng hiệu quả vì các đoạn mã thường tái sử dụng trong các ứng dụng khác nhau. Điều này đặc biệt đúng đối với bộ hậu xử lý.

Hình 6.8: Kiến trúc mô phỏng

Mục đích của bộ tiền xử lý là định rõ tất cả các tham số cần thiết để định nghĩa hệ thống (các tham số hệ thống) và thiết lập các tham số nhằm quản lý việc thực hiện mô phỏng (các tham số bên trong). Ví dụ các tham số hệ thống như: bậc bộ lọc, loại bộ lọc, băng thông, khuếch đại, tỷ lệ mã, tỷ số trải phổ, tần số sóng mang, tốc độ bit và SNR. Các tham số bên trong gồm tần số lấy mẫu, thời gian ổn định và số mẫu được xử lý. Ngoài ra dữ liệu được qua bộ hậu xử lý để phân tích, hoặc để vẽ, nó phải được lưu giữ bởi mô phỏng, vì vậy cần có các file dữ liệu được xác định bởi bộ hậu xử lý phải. Ví dụ, véc tơ các mẫu dạng sóng, véc tơ thời điểm mẫu (địa chỉ mẫu) được sử dụng trong bộ hậu xử lý để hiệu chỉnh trục thời gian của đồ thị tín hiệu. Một khi mọi thông tin cần thiết được định rõ trong bộ tiền xử lý thì dữ liệu được tạo ra bên trong bộ tiền xử lý thường được ghi vào một file sao cho khả dụng đối với phương tiện mô phỏng và bộ hậu xử lý. Trong các ví dụ Matlab, dữ liệu bộ tiền xử lý thường được viết lên không gian làm việc Matlab.

Phương tiện mô phỏng sẽ đọc dữ liệu trong các file (được tạo bởi bộ tiền xử lý) và thực hiện mô phỏng. Tất nhiên, mục đích của mô phỏng là tạo dữ liệu để giám sát bằng cách sử dụng bộ hậu xử lý. Điều này có thể là lấy dạng các số (SNR, BER, độ lợi mã,...), hoặc véc tơ các giá trị mẫu để xử lý bổ sung bởi bộ hậu xử lý. Thông tin được tạo bởi phương tiện mô phỏng được lưu trong các file, và đi qua bộ hậu xử lý. Trong trường hợp Matlab thường chuyển dữ liệu được tạo bởi phương tiện mô phỏng vào không gian làm việc Matlab.

Bộ hậu xử lý lấy dữ liệu được tạo bởi phương tiện mô phỏng, tạo các sản phẩm mô phỏng cuối cùng cần thiết. Chúng gồm đồ thị dạng sóng, chùm tín hiệu, BER là hàm của E_b/N_0, mật độ phổ công suất, biểu đồ mắt, biểu đồ thống kê... Chi tiết hóa hậu xử lý được xét trong chương 8.

Bộ hậu xử lý cần có mức tín hiệu thoả đáng để hỗ trợ đồ hoạ. Đây là một lý do dùng Matlab cho các ứng dụng trình bày ở đây. Các bộ hậu xử lý hữu hiệu nhất khi chúng có điều khiển menu, tạo ra xử lý tín hiệu khác nhau và các lựa chọn hiển thị. Mô phỏng trong chương này là khá đơn giản như bộ tiền xử lý và bộ hậu xử lý, nhằm giải thích vai trò của mỗi phần tử mô phỏng.

Bộ tiền xử lý, phương tiện mô phỏng và bộ hậu xử lý có thể được viết ở nhiều ngôn ngữ khác nhau. Dự án gần đây của các tác giả sử dụng Visual Basic cho bộ tiền xử lý, C++ cho phương tiện mô phỏng và Matlab cho bộ hậu xử lý. Matlab thường được lựa chọn để triển khai bộ hậu xử lý vì thư viện đồ hoạ phong phú của Matlab.

6.4.2. Mô phỏng

Mô hình mô phỏng là dễ hiểu ngoại trừ có lẽ đối với bộ lọc vòng. Vì hàm truyền đạt bộ lọc vòng như định nghĩa trong (6.25) không phải là hàm quy cách (hàm quy cách là hàm trong đó bậc của đa thức mẫu số lớn hơn bậc của đa thức tử số ít nhất là một), thực hiện chia ta được:

$$F(s) = 1 + \frac{(1-\lambda)a}{s+\lambda a} = 1 + F_1(s) \tag{6.33}$$

Vì vậy, có thể thực thi hóa *F(s)* là sự kết hợp hai hàm truyền đạt, hàm thứ nhất là một hằng số và hàm thứ hai là F1(s). Rõ ràng:

$$F_1(s) = \frac{(1-\lambda)a}{s+\lambda a} = \frac{Y_1(s)}{X_1(s)} \tag{6.34}$$

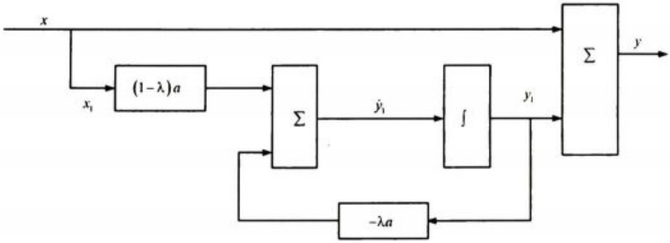
Trong đó $X_1(s)$ và $Y_1(s)$ là biến đổi Laplace đầu vào/ra của bộ lọc con $F_1(s)$. Nhân chéo ta có:

$$sY_1(s) = (1-\lambda)a\,X_1(s) - \lambda a\,Y_1(s) \tag{6.35}$$

Trong miền thời gian nó là:

$$\frac{dy_1}{dt} = (1-\lambda)a\,x_1(t) - \lambda a\,y_1(t) \tag{6.36}$$

Theo đó, ta có bộ lọc vòng được minh hoạ trong hình 6.9.



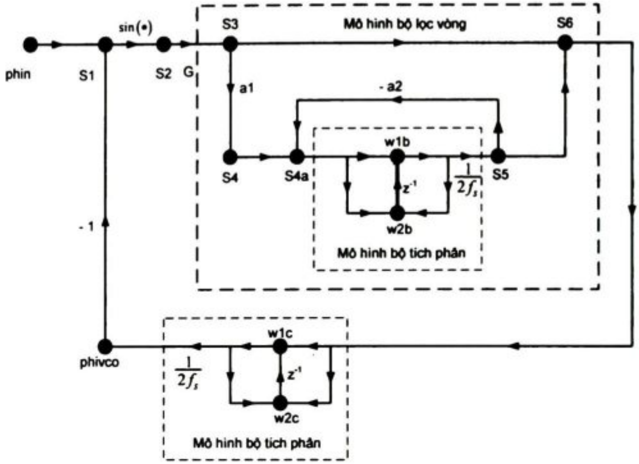
Hình 6.9: Mô hình bộ lọc vòng

Tiếp theo, ta triển khai mã chương trình mô phỏng cho PLL bậc hai, ta triển khai *biểu đồ tiến trình tín hiệu* cho hệ thống và chỉ rõ những điểm trên biểu đồ tiến trình, định nghĩa các tín hiệu. Biểu đồ tiến trình tín hiệu được minh hoạ trong hình 6.10. Lưu ý rằng mô hình bộ lọc vòng tuân theo hình 6.9 với:

$$a_1 = (1 - \lambda)a \tag{6.37}$$

$$a_2 = \lambda a \tag{6.38}$$

Những điểm tại đó các tín hiệu được định nghĩa trong mã mô phỏng được biểu diễn bởi các chấm đen to trên biểu đồ tiến trình tín hiệu. Lưu ý rằng ta đang sử dụng tích **phân hình** thang như đã triển khai trong chương trước.



Hình 6.10: Biểu đồ thị tiến trình tín hiệu cho PLL bậc hai

Mã chương trình mô phỏng trực tiếp tuân theo biểu đồ tiến trình tín hiệu. Mỗi giá trị mẫu được tính toán trong vòng mô phỏng chứa thông tin nhận dạng tương ứng với nhận dạng được định nghĩa trong biểu đồ tiến trình tín hiệu. Ví dụ, dòng mã:

s2 = **sin**(s1)

Định nghĩa tính phi tuyến hình **sin** biểu diễn cho bộ tách pha, trong đó s1 và s2 được xác định trong biểu đồ tiến trình tín hiệu. Tất cả các dòng mã khác trong vòng mô phỏng theo cách tương tự. Vì vậy mã chương trình Matlab để thực hiện vòng mô phỏng chính là:

```
% = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
% Bắt đầu vòng lặp mô phỏng
% = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = = =
for i = 1:npts
 s1 = phin(i) - phivco;  % lỗi pha
 s2 = sin(s1);       % Bộ tách pha hình sin
 s3 = G*s2;
 s4 = a1*s3;
 s4a = s4 - a2*s5; % Đầu vào bộ tích phân bộ lọc vòng
```

```
    w1b = s4a + w2b;   % Bộ tích phân bộ lọc (bước 1)
    w2b = s4a + w1b;   % Bộ tích phân bộ lọc (bước 2)
    s5 = w1b/twofs;    % Tạo đầu ra bộ lọc
    s6 = s3 + s5;      % Đầu vào bộ tích phân VCO
    w1c = s6 + w2c;    % Bộ tích phân VCO (bước 1)
    w2c = s6 + w1c;    % Bộ tích phân VCO (bước 2)
    phivco = w1c/twofs;      % Tạo đầu ra VCO
  end
```

Hằng số **twofs** là gấp hai lần tần số lấy mẫu, được cần đến đối với tích phân hình thang và được tính toán bên ngoài vòng mô phỏng.

Mã chương trình mô phỏng đầy đủ cho PLL bậc hai được cho bởi **NVD6_PLLsim.m** ở Phụ lục 6A. Lưu ý rằng, mã được cho trong Phụ lục 6A là khác chút ít với mã được cho ở đây vì các véc tơ được triển khai cho pha đầu vào, lỗi pha và tần số VCO. Những véc tơ này được yêu cầu bởi bộ hậu xử lý để vẽ các dạng sóng. Mã Matlab cho bộ tiền xử lý ở file **NVD6_PLLpre.m** và bộ hậu xử lý được cho ở file **NVD6_PLLpost.m** cùng với các chương trình con kèm theo có trong Phụ lục 6A. Bộ hậu xử lý được thiết kế theo kiểu điều khiển menu. Các bộ hậu xử lý điều khiển menu thường nằm trong các gói mô phỏng.

6.4.3 Kết quả mô phỏng

Giả sử rằng độ lệch tần số đầu vào PLL là bước nhảy đơn vị tại thời điểm t_0. Vì vậy:

$$\frac{d\phi}{dt} = 2\pi f_\Delta \underbrace{u(t - t_0)}_{\text{bước nhảy đơn vị}} \tag{6.39}$$

Với các tham số PLL dưới đây:

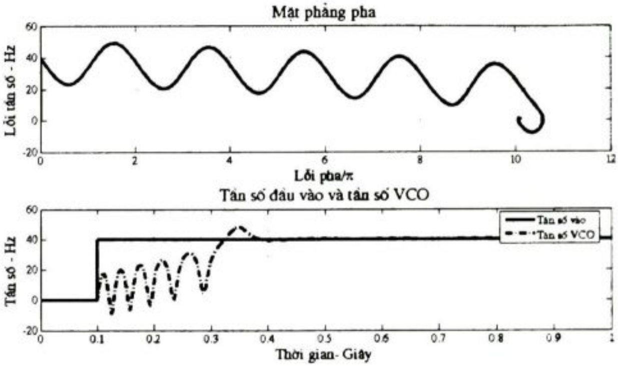
Bước nhảy tần số,	f_Δ	=	40 Hz	
Tần số vòng tự nhiên,	$\omega_n/2\pi$	=	10 Hz	
Độ dịch điểm cực tương đối,	λ	=	0,1	
Hệ số suy giảm vòng,	ς	=	$1/\sqrt{2}$	(6.40)
Tần số lấy mẫu,	f_s	=	5000 Hz	
	T_{stop}	=	0,8 s	

Sử dụng bộ tiền xử lý để nhập các tham số này.

Các lựa chọn **menu** trong bộ hậu xử lý **NVD6_post.m** (Phụ lục 6A) cho phép dễ dàng khảo sát một số các mục quan tâm. Chúng gồm tần số đầu vào và tần số VCO, lỗi tần số và mặt phẳng pha. Ở đây ta chỉ minh hoạ hai đồ thị được tạo bởi bộ hậu xử lý. Mặt phẳng pha được minh hoạ trong nửa trên của hình 6.11. Tần số đầu vào (bước nhảy đơn vị là 40 Hz tại $t_0 = 0{,}08$) và tần số VCO (dạng sóng dao động) được minh hoạ trong nửa dưới của hình 6.11. Cần lưu ý rằng, ta đang xét các mô hình thông thấp. Vì vậy tần số đầu vào và tần số VCO thực ra là độ lệch tần đầu vào và độ lệch tần VCO so với tần số sóng mang chuẩn hóa f_c.

Từ mặt phẳng pha, trượt 5 chu kỳ và lỗi trạng thái ổn định là lớn hơn 10π. Từ hình 6.11 cho thấy bước nhảy tần số đầu vào là 40 Hz. Tuy nhiên, tần số VCO dao động qua 5 chu kỳ và

sau đó đạt được khoá pha ở chu kỳ thứ 6. Đặc điểm này gọi là *"trượt tuần hoàn"* và là các đặc trưng của các bộ đồng bộ phi tuyến khi thay đổi tần số đầu vào vượt quá tần số tự nhiên.



Hình 6.11: Các kết quả mô phỏng

6.4.4. Nguồn lỗi trong mô phỏng

Có một số nguồn lỗi xuất hiện trong mô phỏng. Những nguồn lỗi này đã được đề cập trong chương 1 là do: *(i)* việc ánh xạ thiết bị vật lý sang mô hình phân tích; *(ii)* ánh xạ mô hình phân tích sang mô hình mô phỏng. Dưới đây ta đề cập ngắn gọn.

Mô hình giải tích

Khi triển khai mô hình phân tích cho thiết bị ta thực hiện một số giả định. Một cách điển hình, những xấp xỉ này bao gồm việc ý tưởng hóa các thành phần vòng và có thể là không hợp lệ nếu muốn có được các mô phỏng thiết bị vật lý với độ chính xác cao. Để các kết quả mô phỏng với độ chính xác cao cần phải đo kiểm để xác định các mô hình thành phần đủ chính xác. Những hạn chế khác có thể cũng đưa các tín hiệu vào trong vòng mà không được tính trong mô hình phân tích cơ bản.

Mô hình mô phỏng

Tồn tại một số nguồn lỗi quan trọng thuộc về quá trình ánh xạ từ mô hình giải tích sang mô hình mô phỏng. Ví dụ, trong khi hệ thống vật lý xử lý các tín hiệu liên tục theo thời gian thì mô phỏng xử lý các mẫu tín hiệu đó. Tất nhiên, lấy mẫu dẫn đến xuất hiện các lỗi, cần tính đến các lỗi lượng tử. Ngoài ra các bộ lọc tương tự trong phần cứng (hoặc mô hình giải tích) phải được thế bởi bộ lọc số trong mô hình mô phỏng. Biết rằng, bộ lọc số không bao giờ có cùng đáp ứng biên độ và pha như bộ lọc tương tự. Các lỗi do chồng phổ phải được xét đến.

Một nguồn lỗi quan trọng xuất hiện trong tất cả các hệ thống có hồi tiếp. Ví dụ, đầu ra của bộ tách pha trong PLL tương tự (phần cứng) được định nghĩa là:

$$e_d(t) = \sin\left[\phi(t) - \theta(t)\right] \tag{6.41}$$

Mô hình rời rạc theo thời gian cho đầu ra này sẽ là:

$$e_d(nT) = \sin\left[\phi(nT) - \theta(nT)\right] \tag{6.42}$$

Trong đó $\phi(nT)$ là đầu vào "*hiện thời*" của PLL. Tuy nhiên, không có sẵn độ lệch pha đầu ra $\theta(nT)$ của VCO vì cần phải có $e_d(nT)$ để tính toán nó. "*Sự bế tắc*" tính toán do tính phụ thuộc lẫn nhau giữa $e_d(nT)$ và $\theta(nT)$ được xử lý bằng cách sử dụng giá trị tính toán độ lệch pha trước đó $\theta((n-1)T)$ của VCO để tính $e_d(nT)$. Nói cách khác, mô hình được định nghĩa bởi (6.42) được thay bằng:

$$e_d(nT) = \sin\left[\phi(nT) - \underbrace{\theta((n-1)T)}\right] \tag{6.43}$$

Theo đó, đã gây ra trễ một mẫu trong vòng hồi tiếp (không có trong mô hình vật lý hoặc trong mô hình phân tích). Nếu kích thước bước mô phỏng nhỏ (tần số lấy mẫu cao) thì trễ một mẫu có thể không ảnh hưởng đến tính chính xác của mô phỏng, nhưng không bỏ qua được đối với các kích thước bước mô phỏng thực tế (tần số lấy mẫu thực tế không đủ lớn). Ảnh hưởng trễ một mẫu quy vào dịch pha tuyến tính (trễ thời gian không đổi) trong hàm truyền đạt vòng hở. Điều này làm giảm độ dự trữ pha của hệ thống và nếu kích thước bước mô phỏng đủ lớn thì có thể đẩy hệ thống vào trạng thái bất ổn định.

Nếu mô phỏng là để có mức chính xác cho trước, thì các mô hình mô phỏng phải dựa trên các mô hình giải tích ít nhất là có cùng mức độ chính xác. Nếu thiết bị vật lý đang được mô hình tỏ ra thay đổi đáng kể thì các mô hình mô phỏng cần phải dựa vào dữ liệu đo (được kiểm định từ dụng cụ hiệu chuẩn chính xác). Các bộ khuếch đại phi tuyến thường rơi vào loại này, là thiết bị chứa các thành phần lão hóa. Ví dụ khác là các mô hình cho các kênh đa đường.

Ở đây ta chỉ đề cập một vài nguồn lỗi, nhưng chúng rất quan trọng, là mô hình giải tích bắt giữ một phần đặc tính (hy vọng đó là các đặc tính quan trọng nhất) của các thiết bị vật lý và chấp nhận mắc lỗi phụ khi chuyển mô hình giải tích sang mô hình mô phỏng, mức độ gây lỗi phụ thuộc vào mô hình (giải tích) hệ thống và phương pháp luận mô phỏng. Người dùng mô phỏng phải nhận ra những nguồn lỗi này và đảm bảo lỗi tích lũy đủ nhỏ.

6.5. Dùng mô phỏng để giải phương trình vi phân

Khi máy tính số chưa được dùng phổ biến, máy tính tương tự cho ta công cụ để giải các phương trình vi phân. Máy tính tương tự đã chứng tỏ hữu hiệu nhất đối với các phương trình thay đổi theo thời gian phi tuyến trong đó không tồn tại các kỹ thuật tìm nghiệm giải tích tổng quát. Vì kỹ thuật máy tính tương tự có thể được áp dụng cho các hệ thống phi tuyến hoặc thay đổi theo thời gian hoặc cả hai, nên các máy tính tương tự đã được sử dụng rộng rãi. Trong khi "nghiệm máy tính tương tự" không có dạng của một phương trình (là dạng tiêu chuẩn để biểu diễn nghiệm cho phương trình vi phân) thì nó cho phép dễ dàng vẽ các dạng sóng xuất hiện tại các thời điểm khác nhau trong hệ thống. Dẫn đến dễ dàng nghiên cứu tham số vì vậy cho phép hiểu biết sâu sắc về các đặc tính hoạt động của các hệ thống phức tạp bằng việc sử dụng máy

tính tương tự. Không lấy làm ngạc nhiên về máy tính tương tự đã triển khai thành một công cụ mô phỏng mạnh mẽ.

Tuy máy tính tương tự là công cụ hữu ích nhưng việc sử dụng nó không hề đơn đơn giản. Thành phần cơ bản của máy tính tương tự là bộ khuếch đại thuật toán. Đặc biệt, những bộ khuếch đại thuật toán đó sử dụng trong suốt những ngày, phải chịu trôi điểm làm việc. Vì vậy thường xuyên phải hiệu chuẩn. Ngoài ra việc thực hiện nhân và chia là khó khăn. Các máy tính tương tự chỉ có thể xử lý các tín hiệu có băng thông hạn chế dẫn đến thường xuyên phải lấy tỉ lệ về tần số và thời gian. Khi máy tính số trở nên phổ cập cùng với tính thân thiện trong sử dụng thì máy tính tương tự không được dùng nữa. Tuy nhiên, nhiều phương pháp luận đã được triển khai cho các máy tính tương tự lại có khả năng ứng dụng để mô phỏng sử dụng các máy tính số. Vì vậy kiến thức cơ bản đã triển khai cho các máy tính tương tự mang lại một tập hợp các kỹ thuật mô phỏng quan trọng có khả năng ứng dụng cho các máy tính số. Ví dụ, một trong các chương trình mô phỏng đầu tiên, CSMP (chương trình mô hình hóa hệ thống liên tục) được triển khai bởi IBM đối với dòng máy tính hệ 360, là mô phỏng máy tính số cơ bản của máy tính tương tự. Khi máy tính cá nhân đã phát triển thành công cụ hữu ích cho các nhà khoa học và các kỹ sư thì các phiên bản dựa trên PC khác nhau của CSMP đã được phát triển và một vài số chúng vẫn còn sử dụng cho tới nay. Một trong những chương trình mô phỏng đầu tiên hướng vào các hệ thống truyền thông, TOPSIM dựa trên CSMP.

Sơ đồ mô phỏng

Bước đầu tiên để giải phương trình vi phân sử dụng kỹ thuật máy tính tương tự để triển khai sơ đồ mô phỏng máy tính tương tự. Quá trình này là dễ hiểu. Ví dụ đơn giản, giả sử hệ thống được định nghĩa bởi phương trình sau:

$$\frac{d^2y}{dt^2} + a\frac{dy}{dt} + by = x(t) \tag{6.44}$$

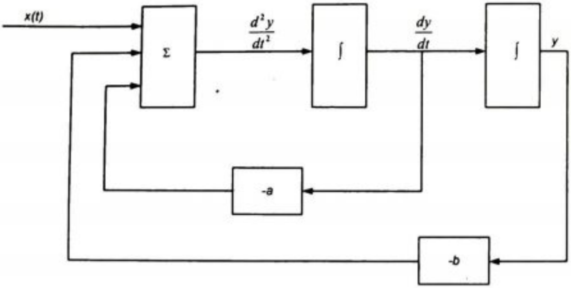
Trong đó a và b nhận giá trị hằng số. Giải phương trình vi phân cho $\frac{d^2y}{dt^2}$ được:

$$\frac{d^2y}{dt^2} = x(t) - a\frac{dy}{dt} - by \tag{6.45}$$

Lấy tích phân d^2y/dt^2 để được dy/dt và lấy tích phân dy/dt để được $y(t)$. Sau đó những đại lượng này được nhân, hoặc được thao tác bằng các hằng số phù hợp hoặc các hàm để tạo các số hạng cần thiết nhằm biểu diễn các số hạng khác nhau trong (6.45). Kết hợp chúng lại tạo phương trình vi phân phù hợp và sơ đồ mô phỏng máy tính tương tự cho phương trình vi phân này được cho ở hình 6.12. Trong khi hình 6.12 đã triển khai cho (6.45), là phương trình tuyến tính đơn giản với các hệ số không đổi và có khả năng giải được bằng cách sử dụng nhiều phương pháp khác nhau thì cái hay của kỹ thuật máy tính tương tự là khả năng ứng dụng vào các hệ thống phi tuyến và thay đổi theo thời gian với bậc bất kỳ.

Ví dụ hệ thống vừa phi tuyến vừa thay đổi theo thời gian, giả sử rằng hệ thống được định nghĩa bởi phương trình vi phân sau:

$$\frac{d^3y}{dt^3} + a(t)y(t)\frac{d^2y}{dt^2} + b(t)y^2(t)\frac{dy}{dt} + c(t)y(t) = x(t) \tag{6.46}$$

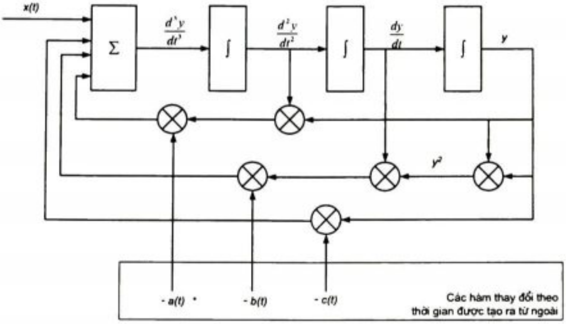


Hình 6.12: Sơ đồ mô phỏng cho hệ thống tuyến tính bất biến theo thời gian bậc hai

Giải tìm d^3y/dt^3 được:

$$\frac{d^3y}{dt^3} = x(t) - a(t)y(t)\frac{d^2y}{dt^2} - b(t)y^2(t)\frac{dy}{dt} - c(t)y(t) \tag{6.47}$$

Từ đó có sơ đồ mô phỏng được cho ở hình 6.13. Các hệ số thay đổi theo thời gian a(t), b(t) và c(t) có thể được định rõ bởi các phương trình phù hợp hoặc có thể là các file của dữ liệu đo được tập hợp từ thực nghiệm.



Hình 6.13: Sơ đồ mô phỏng cho hệ thống bậc ba phi tuyến thay đổi theo thời gian

Nhắc lại PLL

Tại đây ta xét mô phỏng PLL sử dụng giải pháp phương trình vi phân.

Trước hết là rút ra phương trình vi phân. Từ hình 6.3 ta có:

$$E_{vco}(s) = G.F(s).E_d(s) \tag{6.48}$$

Vì VCO trong mô hình PLL được minh hoạ trong hình 6.3 được biểu diễn bởi một bộ tích phân, nên ta có:

$$\Theta(s) = \frac{1}{s} E_{vco}(s) \tag{6.49}$$

Thay (6.48) vào (6.49) và sử dụng (6.25) cho bộ lọc vòng tạo ra:

$$\Theta(s) = \frac{G}{s}\left(\frac{s+a}{s+\lambda a}\right)E_d(s) \tag{6.50}$$

Đơn giản hóa bài toán bằng cách cho rằng vòng bậc hai *"lý tưởng"* khi $\lambda = 0$, dẫn đến:

$$s^2\Theta(s) = s.G.E_d(s) + G.a.E_d(s) \tag{6.51}$$

Vì nhân với s là tương đương với vi phân trong miền thời gian nên ta có phương trình vi phân:

$$\frac{d^2\theta}{dt^2} = G.\frac{d}{dt}e_d(t) + G.a.e_d(t) \tag{6.52}$$

Theo định nghĩa:

$$e_d(t) = \sin\psi(t) \tag{6.53}$$

$$\frac{d^2\psi}{dt^2} = \frac{d^2\phi}{dt^2} - \frac{d^2\theta}{dt^2} \tag{6.54}$$

Phương trình (6.52) được viết là:

$$\frac{d^2\psi}{dt^2} + G\frac{d}{dt}\{\sin\psi(t)\} + G.a.\sin\psi(t) = \frac{d^2\phi}{dt^2} \tag{6.55}$$

Trong đó $\psi(t)$ là lỗi pha PLL và $\phi(t)$ là độ lệch pha của tín hiệu đầu vào. Trước khi triển khai sơ đồ mô phỏng máy tính tương tự, ta viết phương trình trên ở dạng:

$$\frac{d^2\psi}{dt^2} + G\cos\psi(t)\frac{d\psi}{dt} + G.a.\sin\psi(t) = \frac{d^2\phi}{dt^2} \tag{6.56}$$

Sơ đồ mô phỏng cho hệ thống này được minh hoạ trong hình 6.14. Ta mong muốn khảo sát hệ thống này theo sự thay đổi bước nhảy về độ lệch tần số đầu vào tại $t = t_0$. Vì vậy, ta đặt:

$$\frac{d\phi}{dt} = 2\pi f_\Delta u(t - t_0) \tag{6.57}$$

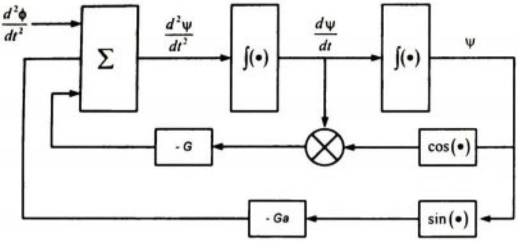
Dẫn đến:

$$\frac{d^2\phi}{dt^2} = 2\pi f_\Delta \delta(t - t_0) \tag{6.58}$$

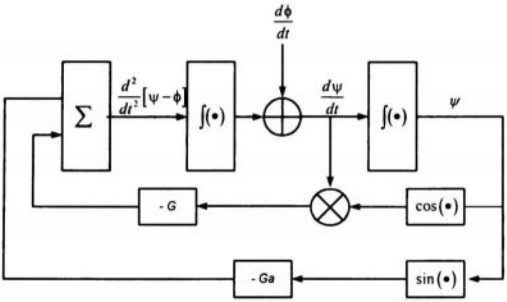
Vì vậy đầu vào được chỉ ra trong hình 6.14 *là một xung kim*.

Ta thường mong muốn đặt đầu vào tại một điểm trong sơ đồ mô phỏng sao cho dễ dàng mô phỏng. Ví dụ, đầu vào trong hình 6.14 là một xung kim. Lấy tích phân xung kim và chuyển

nó sang phải bộ tích phân đầu tiên như được minh họa ở hình 6.15 cho phép ta sử dụng hàm bước nhảy được định nghĩa bởi (6.57) như một "đầu vào tương đương". Những biến đổi đơn giản như kiểu này thường làm mô phỏng đơn giản đáng kể.



Hình 6.14: Sơ đồ mô phỏng cho PLL bậc hai hoàn hảo dựa trên phương trình vi phân



Hình 6.15: Sơ đồ mô phỏng có bước tần số làm đầu vào

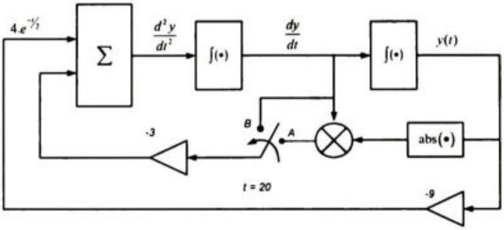
Ví dụ 6.3: Mô phỏng hệ thống vừa có tính phi tuyến vừa có tính thay đổi theo thời gian. Giả sử hệ thống được định nghĩa bởi các phương trình vi phân

$$\frac{d^2y}{dt^2} + \underbrace{3\left|y(t)\right|\frac{dy}{dt}}_{\text{thành phần phi tuyến}} + 9y(t) = 4e^{-t/2} \qquad t < 20 \tag{6.59}$$

$$\frac{d^2y}{dt^2} + 3\frac{dy}{dt} + 9y(t) = 4e^{-t/2} \qquad t \ge 20 \tag{6.60}$$

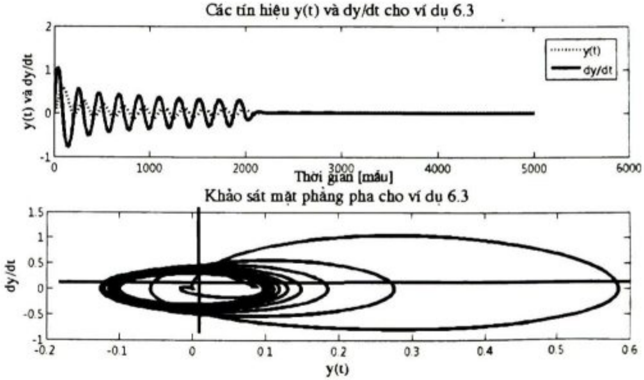
Hệ thống là phi tuyến vì số hạng $\left|y(t)\right|\frac{dy}{dt}$ trong phương trình *định nghĩa* hệ thống khi $t < 20$, và hệ thống thay đổi theo thời gian vì dạng của phương trình vi phân là phụ thuộc thời gian. Hệ thống được đặc trưng hóa bởi hai phương trình vi phân riêng biệt được thực hiện bằng cách chuyển mạch vị trí tại t = 20 giây như được minh họa ở hình 6.16. Ở vị trí A hệ thống là

phi tuyến (6.59), ở vị trí B hệ thống là *tuyến tính* (6.60). Mặt phẳng pha sẽ minh hoạ như mong đợi là: đáp ứng của hệ thống *phi tuyến* là hoàn toàn khác với đáp ứng của hệ thống *tuyến tính*. Như ta đã biết, giải các phương trình vi phân phi tuyến là một nhiệm vụ vô cùng khó khăn vì vậy ta chuyển sang mô phỏng. Chương trình Matlab thực thi hệ thống ví dụ được cho ở file **NVD6_nltvde.m** trong Phụ lục 6A. Lưu ý những nét giống nhau giữa hệ thống này và PLL bậc hai. Vì cả hai hệ thống đều là bậc hai nên cả hai hệ thống này đều phải có 2 bộ tích phân nhưng điểm khác nhau cơ bản giữa chúng là phương trình tại đầu ra của chỗ nối tổng (vị trí làm kín vòng mô phỏng).



Hình 6.16: Minh họa hệ thống phi tuyến thay đổi theo thời gian

Thực hiện chạy chương trình mô phỏng **NVD6_nltvde.m** ở Phụ lục 6A, kết quả là một mặt phẳng pha được minh hoạ trong hình 6.17. Lưu ý rằng *quỹ đạo* pha cả hai trường hợp đều bắt đầu và kết thúc tại gốc tọa độ (0,0). Điểm mà tại đó chuyển mạch lật trạng thái t = 20s được chỉ thị một cách xấp xỉ bởi mũi tên (xem hình 6.17). Ta thấy rằng mặt phẳng pha dẫn tới vòng tròn giới hạn như được chỉ ra bởi sự hội tụ của quỹ đạo pha *tới* vòng khép kín. Hệ thống A hoạt động trong chế độ chu kỳ-hữu hạn là một bộ dao động. Tần số của bộ dao động và dạng sóng của y(t), hoặc đạo hàm của nó có thể được xác định bằng cách vẽ các dạng sóng trong miền thời gian. Dao động sẽ liên tục miễn là chuyển mạch ở vị trí A.



Hình 6.17: Mặt phẳng pha đối với mô phỏng ví dụ 6.3

Chuyển mạch sang vị trí B, dẫn đến hệ thống tuyến tính. Tại thời điểm chuyển sang vị trí B (t = 20 giây) thì đầu vào là không đáng kể (chính xác là $4.e^{-t/2}\Big|_{t=20} = 4.e^{-10}$) và đáp ứng hệ thống xấp xỉ kín thuộc loại hệ thống bậc hai không kích thích, với các điều kiện đầu được cho bởi các giá trị của y(t) và $\frac{dy}{dt}$ tại thời điểm chuyển mạch chuyển từ vị trí A sang vị trí B.

Trong toàn bộ chương này ta đã sử dụng quy tắc tích phân số hình thang để thực hiện xấp xỉ tích phân. Thực hiện vấn đề này là đơn giản, đã được đề cập ở chương trước, thấy rõ là quy tắc hình thang khá chính xác nếu tần số lấy mẫu đủ lớn. Các quy tắc tích phân khác sẽ là chủ đề nghiên cứu sau này khi trình bày nghiên cứu về hệ thống phi tuyến tổng quát hơn (xem chương 12).

Chương 7

TẠO VÀ XỬ LÝ TÍN HIỆU NGẪU NHIÊN

7.1. Mở đầu

Ta đã xét các tín hiệu tất định trong mô phỏng. Trong tất cả các hệ thống truyền thông thực tế, các ảnh hưởng ngẫu nhiên như tạp âm kênh, nhiễu, pha đinh, đều làm suy thoái tín hiệu mang tin. Để mô phỏng chính xác những hệ thống này ở mức dạng sóng cần phải mô hình hóa chính xác các ảnh hưởng ngẫu nhiên. Vì vậy, cần có các thuật toán để tạo các ảnh hưởng ngẫu nhiên này. Khối cơ bản nền tảng là bộ tạo số ngẫu nhiên. Tồn tại nhiều tài liệu được viết cho các bộ tạo số ngẫu nhiên, chương này chỉ tập trung vào việc sử dụng các bộ tạo số ngẫu nhiên trong mô phỏng. Vì vậy, ta sẽ chỉ tập trung nghiên cứu việc tạo các phiên bản mẫu của các dạng sóng ngẫu nhiên (tín hiệu, nhiễu, tạp âm,...) để sử dụng trong các chương trình mô phỏng. Trong môi trường mô phỏng, tất cả các quá trình ngẫu nhiên đều được biểu diễn bằng các chuỗi của các biến ngẫu nhiên. Tạo và kiểm tra các chuỗi ngẫu nhiên này là chủ đề của chương này. Nhiều ngôn ngữ lập trình hữu hiệu để triển khai các chương trình mô phỏng (như Matlab) có sẵn các bộ tạo số ngẫu nhiên trong thư viện. Hiểu những khái niệm về tạo số ngẫu nhiên, giúp hiểu toàn bộ chương trình mô phỏng.

Nói một cách chính xác, các bộ tạo số ngẫu nhiên không thể tạo ra các số hoàn toàn ngẫu nhiên nhưng nó tạo ra các chuỗi số *xuất hiện ngẫu nhiên* trong khoảng thời gian quan sát (mô phỏng), sao cho chúng có thể được dùng để *xấp xỉ* hàm mẫu của một quá trình ngẫu nhiên trong chương trình mô phỏng cụ thể. Do "*xuất hiện ngẫu nhiên*" nghĩa là các chuỗi số được tạo ra trong khoảng thời gian mô phỏng sẽ có các đặc tính cần thiết để mô hình hóa quá trình ngẫu nhiên ở mức độ chính xác cần thiết cho một ứng dụng cụ thể. Những chuỗi số như vậy được coi là các chuỗi số *giả ngẫu nhiên*, mặc dù chúng là tất định nhưng chúng xuất hiện ngẫu nhiên khi được dùng trong ứng dụng cụ thể. Yêu cầu về độ chính xác phụ thuộc vào ứng dụng đó. Ví dụ, nếu ta phải tạo dạng sóng để biểu diễn tạp âm đầu vào PLL, thì yêu cầu độ chính xác để mô hình hóa dạng sóng tạp âm khi SNR đầu vào 50 dB là cao hơn trường hợp 8 dB. Yêu cầu độ chính xác để mô hình hóa thành phần tạp âm trong hệ thống truyền tham số cho trường hợp xác suất lỗi bit 10^{-7} cao hơn so với trường hợp xác suất lỗi bit 10^{-3}.

Trước tiên ta xét việc tạo các hàm mẫu của một quá trình ngẫu nhiên. Ngiên cứu khái niệm dừng trong môi trường mô phỏng. Sau đó, ta xét vắn tắt các mô hình mô phỏng cho các bộ điều chế số. Vì vậy, chương 7 sẽ tập trung chủ yếu vào các vấn đề sau đây:

- Tạo số ngẫu nhiên *không* tương quan phân bố đều trong khoảng (0,1).

- Ánh xạ số ngẫu nhiên *không* tương quan và phân bố đều thành số ngẫu nhiên không tương quan và có hàm mật độ xác suất pdf tuỳ ý (mong muốn).

- Tạo số ngẫu nhiên *không* tương quan và có pdf phân bố Gausơ.

- Tạo số ngẫu nhiên *tương quan* và có pdf phân bố Gausơ.

- Tạo số ngẫu nhiên *tương quan* và có pdf tuỳ ý (mong muốn).

- Sau đó xét vắt tắt việc tạo chuỗi giả ngẫu nhiên PN và một ứng dụng.

Lưu ý rằng: Do tính ngẫu nhiên của tín hiệu và hệ thống truyền thông, nên để triển khai mô phỏng chính xác nhất có thể và hiệu quả cần phải hiểu sâu rộng về các khái niệm như: quá trình ngẫu nhiên, biến ngẫu nhiên, hàm mẫu quá trình ngẫu nhiên, con số thực,... cũng như các tham số đặc trưng... Tồn tại rất nhiều tài liệu cho chủ đề này, để tiện bạn đọc cũng có thể tham khảo Phụ lục **7B1 và 7B2** nếu cần, ở đó trình bày súc tích, cô đọng một cách căn bản về các biến ngẫu nhiên, quá trình ngẫu nhiên, phân loại cũng như nhiều ứng dụng minh họa. Theo quan điểm của tác giả nên đọc phần phụ lục này đặc biệt đối với sinh viên.

7.2. Quá trình dừng và Ergodic

Các quan hệ giữa quá trình ngẫu nhiên, biến ngẫu nhiên, hàm mẫu và số thực (phức) được cho ở hình 7.1. ***Ta nên tham khảo Phụ lục:*** *7B1 và 7B2*.

Nếu cho Q là không gian mẫu và gán cho mỗi kết cục $\xi = \xi_i \in Q$ của một thí nghiệm ngẫu nhiên một hàm thời gian $x(t,\xi_i)$ theo một quy tắc, thì với từng $\xi_i \in Q$, hàm $x(t,\xi_i)$ ký hiệu cho ánh xạ *IR* đến *IR* (hay *C*) theo:

$$x(.,\xi_i): IR \rightarrow IR(hay C), \quad t \mapsto x(t,\xi_i)$$

Các hàm $x(t,\xi_i)$ phụ thuộc vào thời gian được gọi là các thực hiện hay các hàm mẫu. Một quá trình ngẫu nhiên $x(t,\xi)$ là một họ (hay một tập hợp các hàm mẫu $x(t,\xi_i)$, nghĩa là $x(t,\xi) = \{x(t,\xi_i) | \xi_i \in Q\} = \{x(t,\xi_1), x(t,\xi_2),....\}$. Mặt khác, tại một thời điểm $t = t_0 \in IR$, quá trình ngẫu nhiên $x(t_0,\xi)$ chỉ phụ thuộc vào kết cục ξ, là một biến ngẫu nhiên. Vì thế với $t_0 \in IR$, $x(t_0,\xi)$ ký hiệu cho ánh xạ từ Q vào *IR* (hay *C*) theo quy tắc:

$$x(t_0,.): Q \rightarrow IR\ (hay\ C), \quad s \mapsto x(t_0,\xi)$$

Hàm mật độ xác suất của biến ngẫu nhiên $x(t_0,\xi)$ được xác định bởi sự xuất hiện của các kết cục. Vì quá trình ngẫu nhiên là ***một hàm hai biến:*** $t \in IR$ và $\xi \in Q$, nên ký hiệu chính xác là $x(t,\xi)$. Tuy nhiên trong thực tế để đơn giản về ký hiệu thường viết $x(t)$.

Vì vậy, có thể kết luận rằng một quá trình ngẫu nhiên được thể hiện như sau:

1. Nếu t là một biến và ξ là một biến ngẫu nhiên, thì $x(t)$ thể hiện một họ hay một tập hợp các hàm mẫu $x(t,\xi)$.

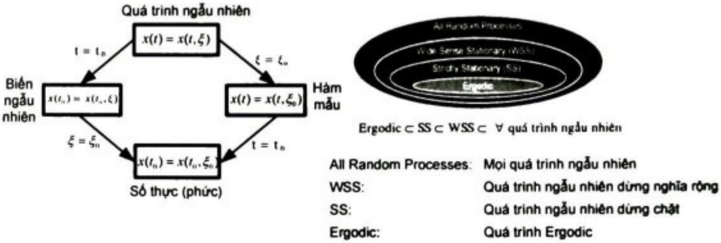
2. Nếu t là một biến và $\xi = \xi_0$ là hằng số, thì $x(t) = x(t,\xi_0)$ là một thực hiện hay một hàm mẫu của quá trình ngẫu nhiên.

3. Nếu $t = t_0$ là một hằng số và ξ là một biến ngẫu nhiên thì $x(t_0)$ cũng là một biến ngẫu nhiên.

4. Nếu cả $t = t_0$ và $\xi = \xi_0$ đều hằng số, thì $x(t_0)$ là một số thực (hoặc phức).

Quan hệ giữa các quá trình ngẫu nhiên, các biến ngẫu nhiên, các hàm mẫu và các số thực (phức) được cho ở hình 7.1.

Khi mô phỏng hệ thống truyền thông, các hàm mẫu được tạo ra để biểu diễn tín hiệu, tạp âm, nhiễu thường được coi là ergodic. Cần có giả định này vì ta xử lý các mẫu dạng sóng đi qua hệ thống một cách liên tiếp và, tại mỗi điểm trong hệ thống có một dạng sóng (hàm mẫu). Ta giả định rằng dạng sóng được xử lý bằng mô phỏng là bộ phận điển hình của toàn bộ được định nghĩa bởi mô hình thống kê cơ bản. Các đại lượng thống kê khác nhau như: các moment, SNR, BER sẽ được tính toán như là các đại lượng trung bình thời gian. Khi so sánh các kết quả mô phỏng với các kết quả lý thuyết tương ứng, thường giả định các trung bình thời gian được tính bởi mô phỏng là tương đương với các trung bình toàn bộ. Kết quả là, có một giả định rằng các quá trình ngẫu nhiên cơ bản là các quá trình ***ergodic***.



Hình 7.1: Quan hệ giữa các quá trình ngẫu nhiên, các biến ngẫu nhiên, các hàm mẫu và các số thực (phức)

Các quá trình ergodic luôn luôn là các quá trình dừng. Vì vậy, các hàm mẫu được tạo ra trong mô phỏng luôn luôn được coi là các thành phần của một quá trình ngẫu nhiên dừng. Từ lý thuyết quá trình ngẫu nhiên cơ bản định nghĩa về vấn đề dừng: nghĩa là tất cả các đại lượng thống kê đều độc lập (không phụ thuộc) vào gốc thời gian. Để được rõ, ta xét một số ví dụ đơn giản sau:

Ví dụ 7.1.1: *Trường hợp thứ nhất*: Giả sử các hàm mẫu của một quá trình ngẫu nhiên được định nghĩa là:

$$x\left(t,\xi_i\right) = A\cos\left(2\pi ft + \phi_i\right) \tag{7.1}$$

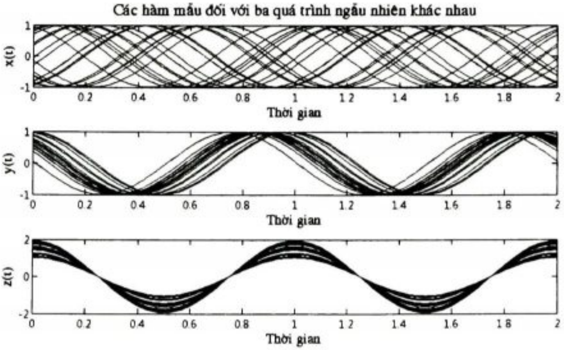
Trong đó ξ_i là một kết cục trong không gian mẫu của thí nghiệm ngẫu nhiên cơ bản, và mỗi kết cục ξ_i được sắp xếp thành một pha ϕ_i. Ta cũng coi thí nghiệm ngẫu nhiên cơ bản bao gồm việc lấy ra một con số từ một bộ tạo số ngẫu nhiên phân bố đều. Kết quả của việc rút ra này là kết cục $\xi_i = u_i$, trong đó u_i được phân bố đều trong khoảng (0,1). Sau đó u_i được ánh xạ lên pha $\phi_i = ku_i$. Khi A và f không đổi, thì giá trị của ϕ_i xác định dạng sóng. Trong ví dụ này cần lưu ý đến 2 giá trị của k là $k = 2\pi$ và $k = \frac{\pi}{2}$, nếu $k = 2\pi$ thì pha được phân bố đều trong khoảng ($0,2\pi$), nếu $k = \frac{\pi}{2}$ thì pha được phân bố đều trong khoảng $\left(0,\pi/2\right)$.

Trường hợp thứ hai: Giả sử rằng quá trình ngẫu nhiên được mô tả bằng biểu thức:

$$x(t,\xi_i) = A(1+u_i).\cos(2\pi ft) \tag{7.2}$$

Trong trường hợp này biên độ được phân bố đều trong khoảng *(A, 2A)*.

Chương trình Matlab **NVD7_ sinewave.m** dưới đây (có ở Phụ lục 7A) tạo ra 3 tập hàm mẫu của một quá trình ngẫu nhiên. Tập dạng sóng đầu tiên được ký hiệu là *x(t)* tương ứng với (7.1) khi $k = 2\pi$. Tập dạng sóng thứ 2 được ký hiệu là *y(t)* tương ứng với (7.1) khi $k = \pi/2$. Tập dạng sóng thứ 3 được ký hiệu là *z(t)* được định nghĩa bởi (7.2). Tất cả đều có $A = 1$ và $f = 1$. Hai giây (thời gian mô phỏng) dữ liệu và 20 hàm mẫu được tạo ra cho mỗi mô phỏng. Kết quả chạy chương trình được minh họa ở hình 7.2(a).



Hình 7.2(a): Các hàm mẫu cho ba quá trình ngẫu nhiên khác nhau

```
% File: NVD7_sinewave.m
f = 1; % Tần số của hình sin
fs = 100; % Tần số lấy mẫu
t = (0:200)/fs; % vectơ thời gian
for i = 1:20
 x(:,i) = cos(2*pi*f*t+rand(1)*2*pi)';
 y(:,i) = cos(2*pi*f*t+rand(1)*pi/2)';
 z(:,i) = (1+rand(1))*cos(2*pi*f*t)';
end
subplot(3,1,1); plot(t,x,'k'); ylabel('x(t)');
subplot(3,1,2); plot(t,y,'k'); ylabel('y(t)');
subplot(3,1,3); plot(t,z,'k'); ylabel('z(t)');
```

Các trung bình thời gian của tất cả các hàm mẫu *x(t), y(t), và z(t)* đều bằng 0. Ta dễ dàng kiểm chứng bằng cách lấy trung bình toàn bộ *x(t)* xấp xỉ 0 khi được tính tại một số lượng lớn

các điểm t_i, $0 \le t_i \le 2$. *Các trung bình thời gian sẽ hội tụ về 0 khi số các hàm mẫu tiến đến* ∞. Tuy nhiên, đối với *y(t)* thì trung bình toàn bộ xấp xỉ 1 khi *t* lân cận 0,375; 1,875. xấp xỉ -1 khi *t* lân cận 0,375; 1,375 và xấp xỉ 0 khi *t* lân cận 0,125; 0,625; 1,125; 1,625. Đây là một ví dụ về quá trình dừng *tuần hoàn (cyclostationary process)*, trong đó các moment là tuần hoàn (*periodic*). Để được rõ hơn, bạn đọc nên tham khảo phần phụ lục 7B2, ở đó trình bày vắt tắt, súc tích, và đặc biệt trực quan hóa bằng nhiều hình vẽ.

Các hàm mẫu *z(t)* cũng là các hàm mẫu từ quá trình dừng tuần hoàn. Lưu ý rằng, việc lấy mẫu quá trình đó tại $t = 0{,}5k$ tạo ra một biến ngẫu nhiên có trung bình xấp xỉ +1,5 khi *k* chẵn, và xấp xỉ -1,5 khi *k* lẻ.

Trong ví dụ 7.1 ta đã sử dụng bộ tạo số ngẫu nhiên đều **rand** có sẵn trong thư viện Matlab. Trong ví dụ 7.2 ta minh hoạ việc sử dụng bộ tạo số ngẫu nhiên để mô hình hóa bộ điều chế số. Trong phần sau sẽ khảo sát chi tiết các thuật toán thực hiện các bộ tạo số đều.

Ví dụ 7.2: Ta thường cần có các mô hình của các bộ điều chế số. Khối cơ bản cho các bộ điều chế này là hàm **NVD7_random_binary.m**, nó tạo dạng sóng nhị phân có các giá trị +1 và -1. Số các bit được tạo ra và số các mẫu trên bit là các đối số của hàm này. Để dùng hàm này trong các chương trình mô phỏng, phải nhập giá trị cho hai tham số là: số bit (*nbit*) và số mẫu trên bit (*nsamples*) theo dòng lệnh sau:

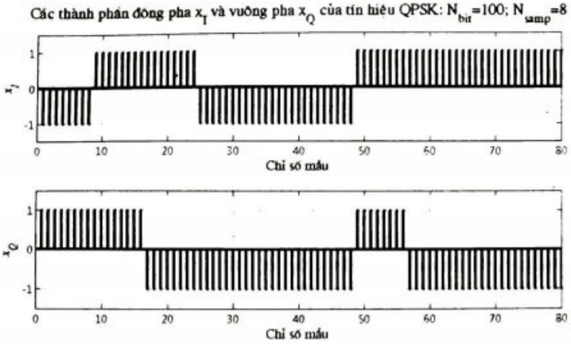
$$\Big[\underbrace{x,\ bits}_{\text{kết quả mô phỏng}}\Big] = \underbrace{NVD7_random_binary}_{\text{Tên hàm cũng là tên file}}\Big(\underbrace{nbit,\ nsamples}_{\text{Các tham số đầu vào (hay đối số) của hàm}}\Big)$$

```
function [ x, bits] = NVD7_random_binary(nbits,nsamples)
% Hàm này tạo dang sóng nhị phân ngẫu nhiên dài nbits được lấy mẫu
  tại tốc độ % nsamples/bit.
x = zeros(1,nbits*nsamples);
bits = round(rand(1,nbits));
for m = 1:nbits
    for n = 1:nsamples
        index = (m-1)*nsamples + n;
        x(1,index) = (-1)^bits(m);
 end
end
```

Hàm **NVD7_random_binary.m** được dùng để mô phỏng một số bộ điều chế số. Ví dụ dùng câu lệnh sau để mô phỏng bộ điều chế QPSK.

$$x = NVD7_random_binary(nbit,nsamples) + i * NVD7_random_binary(nbit,nsamples);$$

Chương trình Matlab **NVD7_ sim2.m** tạo tín hiệu QPSK cho 10 bit với tần số lấy mẫu là 8 mẫu trên bit. Chạy chương trình tạo ra tín hiệu QPSK có các thành phần đồng pha và vuông pha được minh hoạ trong hình 7.2 (b). Lưu ý rằng, ở đây xét tín QPSK băng tần cơ sở.



Hình 7.2(b): Các thành phần đồng pha và vuông pha của tín hiệu QPSK

```
% File: NVD7_sim2
nbits = 10;
nsamples = 8;
x = NVD7_random_binary(nbits,nsamples)+i*NVD7_random_binary(nbits,
    nsamples);
xd = real(x);
xq = imag(x);
% ==================
% Hiển thị kết quả
% ================
subplot(2,1,1)
stem(xd,'.');
grid;
axis([ 0 80 -1.5 1.5]);
xlabel('Chỉ số mẫu','fontname','.vntime','fontsize',12);
ylabel('X_I','fontname','.vntime','fontsize',16)
subplot(2,1,2)
stem(xq,'.');
grid;
axis([ 0 80 -1.5 1.5]);
xlabel('Chỉ số mẫu','fontname','.vntime','fontsize',12);
ylabel('X_Q','fontname','.vntime','fontsize',16);
```

7.3. Bộ tạo số ngẫu nhiên phân bố đều

Biến ngẫu nhiên có hàm mật độ xác suất phân bố đều dễ dàng chuyển thành biến ngẫu nhiên có pdf mong muốn khác (không phải là phân bố đều). Vì vậy, bước đầu tiên trong việc tạo biến ngẫu nhiên có pdf cụ thể là tạo biến ngẫu nhiên phân bố đều trong khoảng (0,1) bằng cách: trước hết tạo ra một chuỗi số (nguyên) giữa 0 và M và sau đó chia mỗi phần tử của chuỗi

cho M. Kỹ thuật thông dụng nhất để thực hiện các bộ tạo số ngẫu nhiên được biết là đồng dư tuyến tính.

7.3.1. Đồng dư tuyến tính

Bộ tạo đồng dư tuyến tính (LCG) được định nghĩa bằng phép toán:

$$x_{i+1} = [ax_i + c]\mathrm{mod}(m) \quad (7.3)$$

Trong đó a được coi là số nhân, c được coi là số gia, và tham số m được gọi là modulus. Tất nhiên, đây là thuật toán tuần tự tất định trong đó các giá trị kế tiếp của x được tạo ra lần lượt. Giá trị khởi đầu của x ký hiệu là x_0 được gọi là số gốc của bộ tạo số. Giả thiết rằng x_0, a, c, m là các số nguyên, thì tất cả các số được tạo ra bởi LCG sẽ là những số nguyên. Tính chất mong muốn tại đầu ra bộ tạo số là nó có chu kỳ dài sao cho tạo được nhiều số nguyên nhất trong chuỗi đầu ra trước khi chuỗi được lặp lại. Với giá trị của m cho trước khi chu kỳ được cực đại hóa ta nói rằng bộ tạo số là *tuần hoàn đầy đủ (full period)*. Ngoài ra, khi ứng dụng vào một chương trình mô phỏng cụ thể đặt các yêu cầu khác lên LCG. Ví dụ ta thường cần có các mẫu x_i và x_{i+1} không tương quan nhau. Hơn nữa, tùy vào ứng dụng cụ thể mà đầu ra LCG phải qua các phép thử thống kê khác. LCG có thể có nhiều dạng khác nhau. Trong phần này, chỉ xét những thuật toán phổ biến nhất.

Kỹ thuật A: Thuật toán đồng dư hỗn hợp

Thuật toán đồng dư tổng quát nhất là thuật toán đồng dư "***hỗn hợp***" trong đó $c \neq 0$. Sở dĩ gọi là hỗn hợp vì cả phép cộng và phép nhân đều được dùng để tính x_{i+1}. Thuật toán tuyến tính hỗn hợp có dạng được cho bởi (7.3):

$$x_{i+1} = [ax_i + c]\mathrm{mod}(m) \quad (7.4)$$

Khi $c \neq 0$ thì bộ tạo số có chu kỳ lớn nhất là m. Đạt được chu kỳ này nếu và chỉ nếu:

- Gia số c là nguyên tố tương đối đối với m. Nói cách khác c và m không có các thừa số nguyên tố chung.
- $a-1$ là bội số của p, trong đó p thể hiện các thừa số nguyên tố của mô đun m.
- $a-1$ là bội số của 4 nếu m là bội số của m.

Ví dụ 7.3: Thiết kế một bộ tạo số đồng dư hỗn hợp có chu kỳ $m = 5000$. Vì:

$$5000 = (2^3)(5^4) \quad (7.5)$$

Ta có thể đảm bảo rằng m và c là nguyên tố tương đối bằng cách đặt c bằng tích của các số nguyên tố chứ không phải 2 và 5. Điều này thoả mãn tính chất thứ nhất. Một trong nhiều khả năng có thể được là đặt:

$$c = (3^2)(7^2) = 1323 \quad (7.6)$$

Bây giờ phải chọn giá trị của a. Tính chất thứ 2 được thoả mãn bằng cách đặt:

$$a - 1 = k_1 p_1 \quad (7.7)$$

$$a - 1 = k_2 p_2 \quad (7.8)$$

Trong đó $p_1 = 2$ *và* $p_2 = 5$ (các thừa số của m) và k_1 & k_2 là các số nguyên tuỳ ý.

Vì 4 là một thừa số của $m = 5000$ nên ta làm thoả mãn chấm thứ 3 bằng cách đặt:

$$a - 1 = 4k_3 \quad (7.9)$$

Trong đó k_3 là số nguyên tuỳ ý. Một sự lựa chọn rõ ràng cho a là đặt:

$$a - 1 = 4.k.p_1.p_2 \quad (7.10)$$

Hoặc:

$$a - 1 = 2.4.5.k = 40k \quad (7.11)$$

Trong đó k là một số nguyên. Với $k = 6$ thì $a = 241$. Vì vậy:

$$x_{i+1} = [241.x_i + 1323]\mathrm{mod}(5000) \quad (7.12)$$

Là một bộ tạo số chu kỳ đầy đủ. Lưu ý rằng, còn có nhiều sự lựa chọn khác cho các tham số để tạo ra bộ tạo số chu kỳ đầy với $m = 5000$.

Ví dụ 7.4: Ví dụ này, ta chỉ ra rằng, LCG được thiết kế trong ví dụ 7.3 thực ra có chu kỳ $m = 5000$. Chương trình Matlab **NVD7_LCGperiod.m** dưới đây (có ở Phụ lục 7A), khi nhập số gốc và chương trình chạy cho đến khi số gốc lại xuất hiện. Nếu n số nguyên được tạo ra và $n > m$ mà không xuất hiện lại số gốc đó thì coi rằng bộ tạo số bị rơi vào một vòng trong đó một chuỗi ngắn được tạo ra một cách lặp.

Chạy chương trình Matlab, chương trình yêu cầu nhập các tham số sau:

```
>> NVD7_LCGperiod.m
Nhập a:          241
Nhập c:          1323
Nhập m:          5000
Nhập số gốc:     1
Kết quả chu kỳ là:  5000.
```

Ta thấy rằng chu kỳ thực sự là 5000 như được mong đợi.

Kỹ thuật B: Thuật toán nhân với mô đun nguyên tố

Bộ tạo số nhân được định nghĩa là:

$$x_{i+1} = [ax_i]\mathrm{mod}(m) \quad (7.13)$$

Là thuật toán hỗn hợp khi $c = 0$. Lưu ý rằng, x_i không thể bằng không khi $c = 0$. Vì vậy, *chu kỳ đầy là m-1 chứ không phải là m* như trường hợp trước. Thuật toán nhân tạo ra chu kỳ đầy nếu:

- m là nguyên tố (thường yêu cầu m lớn)
- a là phần tử *nguyên thuỷ* mod(m)

Biết rằng, số nguyên tố là số chỉ chia hết cho 1 hoặc chính nó. Cần làm sáng tỏ tính chất thứ 2. Ta hiểu a là phần tử nguyên thuỷ *mod(m)* nếu $a^i - 1$ là bộ số của m với $i = m-1$, nhưng không có giá trị i nhỏ hơn. Nói cách khác a là phần tử nguyên thuỷ *mod(m)* nếu:

$$\frac{a^{m-1} - 1}{m} = k \tag{7.14}$$

Và:

$$\frac{a^i - 1}{m} \neq k, \quad i = 1, 2, 3..., m\text{-}2 \tag{7.15}$$

với số nguyên k tuỳ ý.

Kỹ thuật C: Thuật toán nhân với mô đun phi nguyên tố

Quan trọng nhất là trường hợp mođun m không phải là số nguyên tố mà m bằng luỹ thừa của 2. Nói cách khác:

$$x_{i+1} = [ax_i]\mathrm{mod}(2^n), \text{ với n là số nguyên} \tag{7.16}$$

Trường hợp được định nghĩa bởi (7.16), thì chu kỳ lớn nhất là $2^n/4 = 2^{n-2}$. Đạt được chu kỳ này nếu:

- Số nhân a là 3 hoặc 5 mod(8)
- Số gốc x_0 là lẻ.

Vì tích của 2 số lẻ là một số lẻ nên tất cả các giá trị được tạo ra bởi (7.16) là lẻ nếu x_0 là lẻ. Vì vậy, không có các giá trị chẵn của x_i được tạo ra, nó làm giảm chu kỳ đi hai lần. Các số nguyên lẻ được tạo ra bởi (7.16) được phân thành 2 tập, chỉ một tập được tạo ra từ số gốc cho trước. Điều này làm giảm chu kỳ bởi một hệ số 2 khác. Tập các số nguyên lẻ thực ra được tạo ra phụ thuộc vào việc chọn số gốc.

Ưu điểm sử dụng $m = 2^k$ là dùng tràn số nguyên để thực hiện phép toán *mod(m)*, giảm thời gian tính toán. Thực ra mong muốn giảm thời gian tính toán song chương trình không có khả năng chuyển tải dễ dàng.

7.3.2. Kiểm tra bộ tạo số ngẫu nhiên

Phần trước cho ta các công cụ tạo các số giả ngẫu nhiên phân bố đều trong khoảng (0,1). Tại đây ta mới xét chu kỳ của chuỗi số được tạo ra bởi LCG. Ta vừa muốn có chu kỳ dài vừa muốn có các thuộc tính đáp ứng cho một ứng dụng cụ thể. Ít nhất cần có chuỗi số tương quan *delta* (*trắng*). Các yêu cầu khác có thể cần đến cho ứng dụng đó.

Tồn tại nhiều thuật toán để kiểm tra tính ngẫu nhiên của chuỗi. Thuật toán thông dụng nhất là kiểm tra của Chi-square, kiểm tra của Kolômgorov-Simirnov, và kiểm tra phổ. Việc nghiên cứu các thuật toán này nằm ngoài phạm vi của cuốn sách này. Việc kiểm tra phổ được coi là hữu hiệu nhất. Vì vậy, ở đây mô tả vắt tắt việc kiểm tra phổ, được áp dụng vào thuật toán Wichman-Hill.

Nhiều ứng dụng sau đây, thuộc tính quan trọng nhất phải được đáp ứng là: các phần tử của một chuỗi cho trước là *độc lập, hoặc ít nhất là không tương quan*. Phần này, ta xét 2 kiểm tra đơn giản: biểu đồ tán xạ và kiểm tra Durbin-Watson. Sẽ thấy rõ, các tính chất của chuỗi cho trước áp dụng cho chuỗi đầy đủ (chu kỳ đầy đủ). Nếu chỉ sử dụng một phần của chuỗi thì các tính chất của chuỗi đầy đủ không áp dụng được.

Biểu đồ tán xạ

Biểu đồ tán xạ được minh hoạ tốt nhất bằng một ví dụ.

Ví dụ 7.5: *Biểu đồ tán xạ là vẽ x_{i+1} theo x_i* (nghĩa là *x_{i+1} là hàm của x_i*), nó thể hiện phép đo chất lượng bộ tạo số. Trong ví dụ này, ta xét 2 bộ tạo số ngẫu nhiên được định nghĩa bởi:

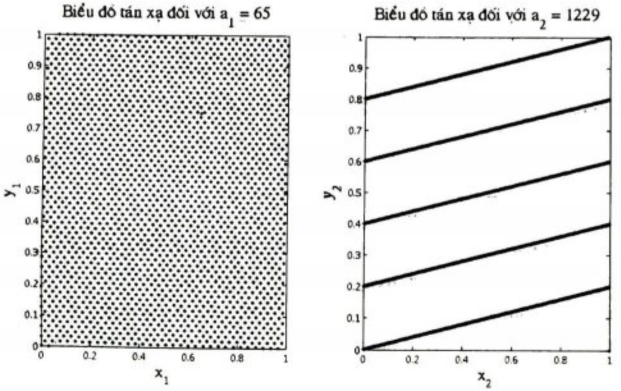
$$x_{i+1} = [65x_i + 1]\text{mod}(2048) \tag{7.17}$$

và:

$$x_{i+1} = [1229x_i + 1]\text{mod}(2048) \tag{7.18}$$

Ứng dụng chương trình **NVD7_LCGperiod.m** trong ví dụ 7.2 cho thấy cả 2 bộ tạo số này đều là bộ tạo số chu kỳ đầy đủ. Mã chương trình Matlab để tạo biểu đồ tán xạ cho các bộ tạo số ngẫu nhiên này được cho bởi file **NVD7_LCGSim1.m** (có trong Phụ lục 7A).

Kết quả chạy chương trình **NVD7_LCGSim1.m** nhận được biểu đồ tán xạ được minh hoạ ở hình 7.3. Nó tìm biểu đồ tán xạ trong đó tất cả các kết hợp tung độ x_{i+1} và toạ độ x_i xảy ra. Trường hợp này biểu đồ phân tán không có cấu trúc. Thấy rõ từ hình 7.3 với a = 65 tạo ra bộ tạo số có tính tương quan nối tiếp nhỏ hơn bộ tạo số có a = 1229.



Hình 7.3: Các biểu đồ tán xạ khi $a_1 = 65$ (bên trái) và $a_2 = 1.229$ (bên phải)

Kiểm tra Durbin-Watson

Kiểm tra Durbin-Watson cho tính độc lập được thực hiện bằng cách tính tham số Durbin:

$$D = \frac{\left(\frac{1}{N}\right)\sum_{n=2}^{N}\left(X[n]-X[n-1]\right)^2}{\left(\frac{1}{N}\right)\sum_{n=1}^{N}X^2[n]} \quad (7.19)$$

Trong đó *X[n]* là biến ngẫu nhiên trung bình 0. Ta sẽ thấy rằng, các giá trị của D ở lân cận 2 tương quan giữa *X[n]* và *X[n-1]* là nhỏ.

Để minh hoạ các tính chất của kiểm tra Durbin-Watson, ta coi *X[n-1]* và *X[n]* là tương quan nhau, và *X[n]* là một quá trình *ergodic*. Để đơn giản hoá ký hiệu ta coi *N* đủ lớn sao cho $N\text{-}1 \approx N$ và viết:

$$D = \frac{E\left\{(X-Y)^2\right\}}{E\left\{X^2\right\}} = \frac{1}{\sigma_X^2}E\left\{(X\text{-}Y)^2\right\} \quad (7.20)$$

Trong đó *X* ký hiệu cho *X[n]*, *Y* ký hiệu cho *X[n-1]*, và *E{.}* ký hiệu cho phép kỳ vọng. Vì giả định *X[n]* và *X[n-1]* tương quan nhau nên ta đặt:

$$Y = \underbrace{\rho X + \sqrt{1-\rho^2}Z}_{\substack{X\&Z:\ \text{Không tương quan nhau} \\ \rho:\ \text{Hệ số tương quan } X\&Y}} \quad (7.21)$$

Trong đó *X* và Z không tương quan nhau và ρ là hệ số tương quan để quan hệ hóa giữa X và *Y*. Lưu ý rằng: *X, Y, Z* đều có cùng phương sai σ^2. Thay (7.21) vào (7.20) được:

$$D = \frac{1}{\sigma^2}E\left\{(1-\rho)^2X^2 - \underbrace{2(1-\rho)\sqrt{1-\rho^2}XZ}_{E[XZ]=0 \text{ vì X \& Z không tương quan}} + (1-\rho^2)Z^2\right\} \quad (7.22)$$

Thành phần ở giữa bằng 0 vì *X* và Z không tương quan nhau và có trung bình 0. Vì *X* và Z có cùng phương sai:

$$D = \frac{(1-\rho)^2\sigma^2 + (1-\rho^2)\sigma^2}{\sigma^2} = 2(1-\rho) \quad (7.23)$$

Vì $-1 \le \rho \le 1$ nên tham số Durbin D thay đổi giữa 0 và 4, với $D = 2$ nếu $\rho = 0$. Các giá trị $D < 2$ ngụ ý tương quan dương, trong khi $D > 2$ ngụ ý các giá trị âm của ρ. Mã chương trình Matlab **NVD7_durbin.m** thực hiện tính toán giá trị của tham số Durbin (có ở Phụ lục 7A).

Ví dụ 7.6: Trong ví dụ này ta tính giá trị *D* cho 2 bộ tạo tạp âm được xét trong ví dụ 7.5. Mã chương trình Matlab được cho bởi **NVD7_ LCDSim2.m** thực hiện bài toán này (có trong Phụ lục 7A).

Chạy chương trình nhận được kết quả:

>> NVD7_ LCDSim2

Giá trị của D_1 là 1,9925 và ρ_1 là 0,0037273.

Giá trị của D_2 là 1,6037 và ρ_2' là 0,19824.

Với $a_1 = 65$ tương quan xấp xỉ bằng 0, trong khi $a_2 = 1,229$ thì tương quan xấp xỉ bằng 0,2. Vì vậy từ kiểm tra Durbin-Watson cho thấy $a_1 = 65$ đưa ra các kết quả tốt hơn $a_2 = 1,229$. Kết quả này là phù hợp với các biểu đồ tán xạ được cho trong hình 7.3.

7.3.3. Các tiêu chuẩn nhỏ nhất

Nhiệm vụ chính là kiểm tra chất lượng LCG bằng cách chỉ ra rằng thỏa mãn được nhiều phép kiểm tra thống kê về tính ngẫu nhiên. Điều này đặc biệt đúng khi chuỗi được tạo ra là dài. Để giải quyết từng phần bài toán này, nhiều thuật toán được xem là các thuật toán *tiêu chuẩn nhỏ nhất*. Thuật toán tiêu chuẩn tối ưu là thuật toán mà

- Chu kỳ đầy đủ
- Thoả mãn tất cả các kiểm tra thống kê khả dụng về tính ngẫu nhiên
- Dễ dàng truyền tải từ máy tính này đến máy tính khác.

Thuật toán như vậy được biết đến và tư liệu hóa một cách chính đáng, nó trở thành *tiêu chuẩn nhỏ nhất*. Thuật toán có thể được dùng với sự tin tưởng mà không cần kiểm tra thêm. Nếu sử dụng thuật toán tiêu chuẩn nhỏ nhất thì không cần lo lắng về *tính chính xác*, nhưng phải đảm bảo rằng thuật toán phải được *thực thi* một cách chính xác trong môi trường tính toán đã cho. Xét dưới góc độ lập trình thì tất cả các số được tạo ra bởi thuật toán đều có thể biểu diễn một cách *duy nhất* (tính duy nhất của các con số).

Tiêu chuẩn nhỏ nhất Lewis, Goodman, và Miller

Tiêu chuẩn này được định nghĩa bởi:

$$x_{i+1} = [16807x_i]\text{mod}(2147483647) \tag{7.24}$$

Trong đó m là số nguyên tố Mersenne (nếu $m = 2^k - 1$ thì m là số nguyên tố Mersenne) $2^{31} - 1$. Giá trị m này đầu tiên được đề xuất bởi Lehmer, người chịu trách nhiệm nhiều trong công việc cơ bản về LCG cách đây hơn nửa thế kỷ. Nó được dùng rộng rãi và được thực hiện dễ dàng đối với số học số nguyên trên các máy tính 32bit, và trong số học dấu phẩy động nếu phần định trị vượt quá 31 bit (từ chương 3 cho thấy, tiêu chuẩn dấu phẩy động IEEE gán 51 bit cho phần định trị).

Thuật toán Wichmann-Hill

Phần trên cho thấy rằng ta mong muốn có các bộ tạo số có chu kỳ dài. Một kỹ thuật hiệu quả để tạo dạng sóng có chu kỳ dài là lấy tổng một số dạng sóng tuần hoàn có chu kỳ khác nhau chút ít. Ví dụ, cos2π(1)t có chu kỳ 1 giây và cos2π(1,0001)t có chu kỳ 10.000/10.001 (nhỏ hơn 1 giây). Dạng sóng tổ hợp được viết ở dạng

$$x(t) = \cos\left(2\pi\underbrace{(10000/10000)}_{1}t\right) + \cos\left(2\pi\underbrace{(10001/10000)}_{1,0001}t\right) \tag{7.25}$$

Nó có chu kỳ 10000 giây (xấp xỉ 2,78 giờ). Trong chu kỳ này, thành phần đầu tiên trải qua 10000 chu kỳ và thành phần thứ 2 trải qua 10001 chu kỳ.

Cũng kỹ thuật đó có thể được áp dụng cho các LCG bằng cách kết hợp một vài bộ tạo số khác nhau, nhưng có các chu kỳ gần như bằng nhau. Thuật toán Wichmann-Hill có lẽ là một ví dụ minh họa tốt nhất về bộ tạo số số kết hợp. Có thể có nhiều thay đổi khác nhau của thuật toán Wichmann-Hill. Thuật toán ban đầu được mô tả trong bài báo của Coates sử dụng 3 bộ tạo số được định nghĩa là:

$$x_{i+1} = (171x_i)\bmod(30269) \tag{7.26}$$

$$y_{i+1} = (170y_i)\bmod(30307) \tag{7.27}$$

$$z_{i+1} = (172z_i)\bmod(30323) \tag{7.28}$$

Thực sự là 3 bộ tạo chu kỳ đầy đủ. Kết hợp ba bộ tạo này cho ta đầu ra:

$$u_i = \left(\frac{x_i}{30269} + \frac{y_i}{30307} + \frac{z_i}{30323}\right)\bmod(1) \tag{7.29}$$

Thuật toán Wichmann-Hill tương đương với LCG nhân có số nhân a và mô đun m như sau:

$$a = 16.555.425.264.690 \tag{7.30}$$

$$m = 30269.30307.30323 \approx 2,7817.10^{13} \tag{7.31}$$

Vì M không phải là số nguyên tố nên chu kỳ ngắn hơn $m - 1$. Người ta đã chỉ ra rằng, chu kỳ xấp xỉ bằng 7.10^{12}, mặc dù nhỏ hơn m nhưng vẫn cực kỳ dài.

Thuật toán Wichmann-Hill, mặc dù khác chút ít về kiến trúc với tiêu chuẩn nhỏ nhất, nhưng được xem là bộ tạo số phân bố đều tiêu chuẩn nhỏ nhất, vì khi mở rộng kiểm tra đã cho thấy rằng nó thoả mãn tất cả các kiểm tra thống kê tiêu chuẩn, và dễ dàng truyền tải từ một máy này tới một máy khác.

7.3.4. Thực hiện trên Matlab

Trước phiên bản Matlab 5, bộ tạo số ngẫu nhiên phân bố đều **rand** trong thư viện Matlab là bộ tạo số tiêu chuẩn nhỏ nhất được định nghĩa bởi (7.24). Bộ tạo số ngẫu nhiên phân bố đều được dùng trong các phiên bản Matlab 5 và 6 dựa trên kỹ thuật được phát triển bởi Marsaglia. Bộ tạo số này có mục đích là tạo ra các số dấu phẩy động chứ không phải là các số nguyên tỉ lệ, được đề cập trong bài báo của Moler. MathWorks khẳng định rằng bộ tạo số này có chu kỳ lớn hơn 2^{1492} và "*khá chính xác*", tạo ra tất cả các số dấu phẩy động giữa *eps* và *1-eps/2*, trong đó hằng số *eps* của Matlab đã được đề cập ở chương 3 là 2^{-52}. Bộ tạo số mới chỉ sử dụng phép cộng và phép trừ. Vì không sử dụng phép nhân hoặc chia nên thuật toán thực hiện nhanh hơn nhiều LCG.

7.3.5. Số gốc và véc-tơ

Vì các ví dụ mô phỏng được trình bày trong cuốn sách dựa trên Matlab nên ta xét vắt tắt cách mà Matlab xử lý các số gốc. Bộ tạo số ngẫu nhiên Matlab "cũ" (trước Matlab 5 và được định nghĩa bởi (7.14)) đã sử dụng một con số gốc. Số ngẫu nhiên mới sử dụng số gốc véc tơ được coi là trạng thái của bộ tạo số. Véc-tơ này gồm 35 phần tử (32 số dấu phẩy động, hai số nguyên, và một cờ) xác định trạng thái của bộ tạo số. Với Matlab 5 hoặc mới hơn, dùng cả 2 bộ

tạo số, trong đó mặc định là bộ tạo số ngẫu nhiên mới. Số ngẫu nhiên cũ được định nghĩa bởi (7.24) sử dụng lệnh RAND('seed',0) hoặc RAND('seed',J). Với mọi lệnh Matlab, ta nên nghiên cứu cẩn thận thông tin được cho bởi lệnh **help**. Ngoài ra, ta nên quan tâm đến các vấn đề sau (thuật ngữ số gốc **seed** được dùng để bao gồm cả các số gốc nguyên và các véc-tơ trạng thái):

- Ta có thể sử dụng cả số gốc mặc định hoặc chỉ rõ số gốc.
- Đóng và mở lại Matlab sẽ thiết lập lại số gốc vào giá trị mặc định. Vì vậy, nếu thực hiện gọi bộ tạo số ngẫu nhiên N lần, sau đó đóng và mở lại Matlab và thực hiện gọi một bộ tạo số ngẫu nhiên nhiều hơn N lần, thì sẽ tạo ra cùng N số trong cả 2 trường hợp. Tính chất này là ưu điểm Matlab, vì nó cho phép tạo lại các chuỗi kết quả giống nhau, hữu ích cho các mục đích kiểm tra.
- Đồng hồ hệ thống được dùng để ngẫu nhiên hóa số gốc khởi đầu.
- Các số gốc được lưu trữ trong bộ đệm và không xuất hiện trên không gian làm việc của Matlab vì vậy khi thực hiện lệnh *clear all* không có ảnh hưởng.

7.4. Ánh xạ biến ngẫu nhiên phân bố đều thành biến ngẫu nhiên có pdf tuỳ ý

Nhiều phương pháp khác nhau đã được triển khai để ánh xạ biến ngẫu nhiên phân bố đều sang biến ngẫu nhiên có pdf không đều. Xảy ra 3 tình huống khác nhau cơ bản sau:

1. *Biết trước phân bố tích lũy đối với biến ngẫu nhiên đích ở dạng kín.* Ta sẽ thấy rằng nếu biết CDF của biến ngẫu nhiên đích ở dạng kín thì có thể sử dụng phương pháp biến đổi ngược (kỹ thuật rất đơn giản).
2. Biết trước pdf của biến ngẫu nhiên đích dạng kín nhưng CDF không được biết ở dạng kín. Biến ngẫu nhiên phân bố Gausơ thuộc loại này. Tồn tại một số phương pháp đặc biệt cho trường hợp này, và ngoài ra có thể dùng các phương pháp loại bỏ.
3. Cả pdf và CDF đều không được biết ở dạng kín. Thường gặp phải tình trạng này khi phải triển khai bộ tạo số ngẫu nhiên để phù hợp hóa với pdf của dữ liệu được tập hợp từ *thực nghiệm.*

Ta khảo sát các kỹ thuật được dùng trong mỗi trường hợp trên.

7.4.1. Phương pháp biến đổi ngược

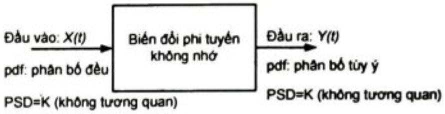
Phương pháp biến đổi ngược cho phép ta chuyển chuỗi ngẫu nhiên không tương quan phân bố đều U thành chuỗi không tương quan (các mẫu độc lập) X có hàm phân bố $F_X(x)$. Việc chuyển đổi dẫn đến sử dụng thiết bị phi tuyến không nhớ như được cho ở hình 7.4. Thực ra thiết bị phi tuyến không nhớ để đảm bảo chuỗi đầu ra không tương quan nhau nếu chuỗi đầu vào không tương quan nhau. Tất nhiên, theo định lý Weiner-Khitchine, một chuỗi các số ngẫu nhiên không tương quan có mật độ phổ công suất PSD không đổi (*trắng*). Kỹ thuật này được thiết lập đơn giản:

$$U = F_X(X) \tag{7.32}$$

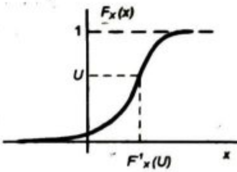
Và tìm x, ta có:

$$X = F_X^{-1}(U) \tag{7.33}$$

Để áp dụng kỹ thuật biến đổi ngược vào việc tìm x cần phải có hàm phân bố $F_X(x)$ ở *dạng kín*.



Hình 7.4: Phương pháp biến đổi ngược



Hình 7.5: Hàm phân bố tích lũy

Dễ dàng thấy rằng, kỹ thuật biến đổi ngược tạo ra biến ngẫu nhiên có hàm phân bố được yêu cầu. Biết rằng, phân bố $F_X(x)$ là hàm không giảm của đối số x như được minh hoạ trong hình 7.5. Theo định nghĩa:

$$F_X(x) = \Pr\{X \le x\} \tag{7.34}$$

Đặt $X = F^{-1}(U)$ và thừa nhận rằng $F_X(x)$ là *đơn điệu* ta được:

$$F_X(x) = \Pr\left\{\underbrace{F^{-1}(U)}_{X} \le x\right\} = \Pr\{U \le F_X(x)\} = F_X(x) \tag{7.35}$$

Là kết quả mong muốn. Ta minh họa phương pháp này thông qua ví dụ đơn giản dưới đây.

Ví dụ 7.7: Biến ngẫu nhiên đều sẽ được chuyển thành biến ngẫu nhiên có phân bố mũ một phía.

$$f_X(x) = \beta e^{-\beta x}u(x), \quad \beta > 0 \tag{7.36}$$

Trong đó $u(x)$ là hàm bước nhảy đơn vị được định nghĩa là:

$$u(x) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases} \tag{7.37}$$

Trước hết là tìm hàm phân bố tích lũy CDF:

$$F_X(x) = \int_0^x \beta e^{-\beta y}dy = 1 - e^{-\beta x} \tag{7.38}$$

Phương trình hóa hàm phân bố với biến ngẫu nhiên đều U theo (7.38) và (7.32) ta được:

$$1 - e^{-\beta X} = U \tag{7.39}$$

Khi tìm X ta có:

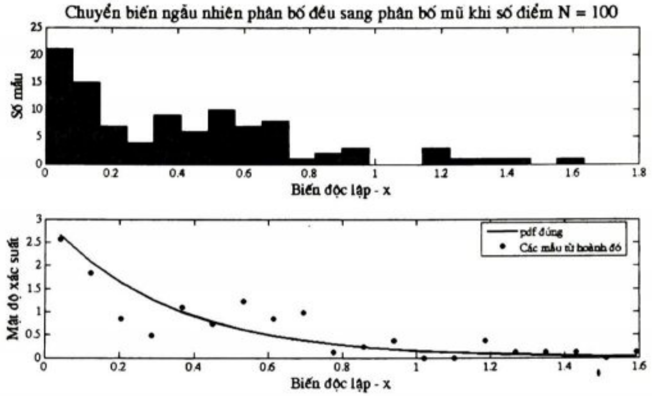
$$e^{-\beta X} = 1 - U \tag{7.40}$$

Vì biến ngẫu nhiên $1-U$ là tương đương với biến ngẫu nhiên U ($Z = U$ và $Z = 1-U$ có cùng hàm mật độ xác suất pdf) nên ta có thể viết lời giải cho X là:

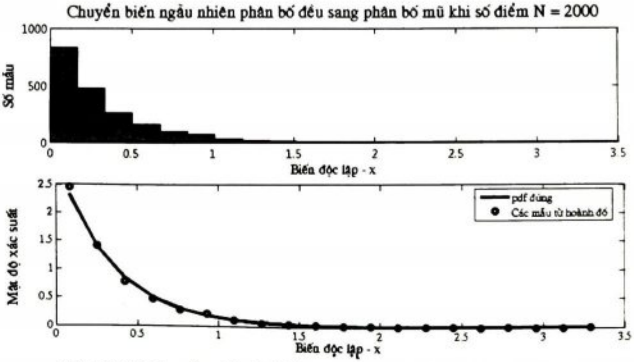
$$X = -\frac{1}{\beta}\ln(U) \tag{7.41}$$

Mã chương trình Matlab chuyển đổi phân bố đều sang phân bố luỹ thừa được cho bởi **NVD7_uni2exp.m** dưới đây (có trong Phụ lục 7A).

Kết quả chạy chương trình với $\beta = 3$ và N = 100 được minh hoạ trong hình 7.6. Phần trên của hình cho thấy hoành đồ (*Histogram*). Phần dưới của hình cho thấy pdf lý thuyết và các giá trị "*thí nghiệm-mô phỏng*" với N = 100 mẫu. Các kết quả tương đối kém khi N = 100 dẫn đến cần thực hiện với nhiều mẫu dữ liệu hơn. Kết quả khi N = 2000 được minh hoạ trong hình 7.7 thấy rõ kết quả được cải thiện đáng kể (thực nghiệm tiệm cận đến lý thuyết).



Hình 7.6: Chuyển phân bố đều thành phân bố mũ khi N = 100



Hình 7.7: Chuyển phân bố đều thành phân bố mũ khi N = 2000

Ví dụ 7.8: Xét biến ngẫu nhiên Rayleigh được mô tả bởi pdf:

$$f_R(r) = \frac{r}{\sigma^2}.e^{-\frac{r^2}{2\sigma^2}}.u(r) \tag{7.42}$$

Trong đó dùng hàm bước nhảy đơn vị *u(r)* để xác định pdf là một phía. CDF được cho bởi:

$$F_R(r) = \int_0^r \frac{y}{\sigma^2} e^{-\frac{y^2}{2\sigma^2}} dy = 1 - e^{-\frac{r^2}{2\sigma^2}} \tag{7.43}$$

Đặt $F_R(R) = U$ ta có:

$$1 - e^{-\frac{R^2}{2\sigma^2}} = U \tag{7.44}$$

Tương đương với:

$$e^{-\frac{R^2}{2\sigma^2}} = U \tag{7.45}$$

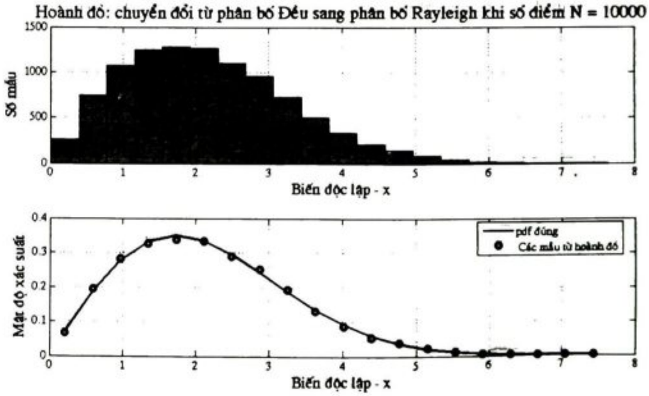
Trong đó một lần nữa ta lại thừa nhận sự tương đương giữa *1 - U* và *U*. Giải tìm *R* ta có:

$$R = \sqrt{-2\sigma^2 \ln(U)} \tag{7.46}$$

Sự chuyển đổi này là bước khởi đầu trong giải thuật Box-Muller là một trong những thuật toán cơ bản để tạo số Gausơ.

Cũng như ở ví dụ 7.7, ta quan tâm thực hiện chuyển đổi và định lượng hiệu năng là *hàm của số điểm được chuyển đổi*.

Chương trình Matlab **NVD7_uni2ray.m** dưới đây (có ở Phụ lục 7A), các kết quả chạy chương trình với $N = 10000$ được cho ở hình 7.8. Bạn đọc nên thực hiện chương trình với các giá trị khác của N và so kết quả với hình 7.8.



Hình 7.8: Chuyển phân bố đều thành phân bố Rayleigh với N = 100000

Hai ví dụ trên đã minh hoạ việc ứng dụng phương pháp biến đổi ngược cho các biến ngẫu nhiên liên tục. Tuy nhiên, kỹ thuật đó có thể áp dụng cho các biến ngẫu nhiên rời rạc. Phương pháp hoành đồ là phiên bản số (dữ liệu rời rạc) của phương pháp biến đổi ngược.

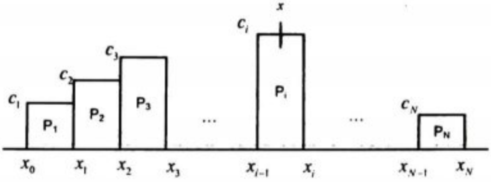
7.4.2. Phương pháp hoành đồ

Hoành đồ (***histogram:*** *hoành đồ - tổ chức đồ - biểu đồ thống kê*) là bộ ước tính hàm mật độ xác suất. Theo đó, giả sử ta có một tập dữ liệu được tập hợp từ thí nghiệm. Trong trường hợp cả pdf và CDF là không được biết, mặc dù pdf có thể được xấp xỉ bởi hoành đồ của dữ liệu. Vấn đề là triển khai một thuật toán để tạo ra một tập các mẫu có pdf xấp xỉ với pdf của dữ liệu thí nghiệm.

Trước hết, ta tạo ra hoành đồ của dữ liệu thí nghiệm. Giả sử hoành đồ được minh hoạ ở hình 7.9 là kết quả. Mỗi khi hoành đồ được tạo ra, thì ta được một xấp xỉ cho pdf và CDF, và vì vậy ta có thể áp dụng phương pháp biến đổi ngược. Kỹ thuật được trình bày ở đây là một sự mở rộng đơn giản của phương pháp biến đổi ngược.

Xác suất mà giá trị mẫu x nằm trong biến thứ i của hoành đồ là:

$$P_i = \Pr\{x_{i-1} < x < x_i\} = c_i(x_i - x_{i-1}) \quad (7.47)$$



Hình 7.9: Hoành đồ của dữ liệu thí nghiệm

CDF được ước lượng tại điểm x ký hiệu là:

$$F_X(x) = F_{i-1} + c_i(x - x_{i-1}) \quad (7.48)$$

Trong đó:

$$F_{i-1} = \Pr\{X \le x_{i-1}\} = \sum_{j=1}^{i-1} P_{i-1} \quad (7.49)$$

Sau đó, ta đặt $F_X(X) = U$, trong đó U là biến ngẫu nhiên đều.

Dẫn đến:

$$F_X(X) = U = F_{i-1} + c_i(X - x_{i-1}) \quad (7.50)$$

Giải tìm X ta được:

$$X = x_{i-1} + \frac{1}{c_i}(U - F_{i-1}) \quad (7.51)$$

Thuật toán cho bộ tạo số theo 3 bước sau:

1. Tạo U bằng cách lấy ra một mẫu từ bộ tạo số ngẫu nhiên tạo ra các số phân bố đều rong khoảng (0,1).

2. Xác định giá trị i sao cho:

$$F_{i-1} < U \le F_i \tag{7.52}$$

Trong đó F_i được xác định theo (7.49).

3. Tạo X theo (7.51) và trả X về chương trình đang gọi.

Độ tin cậy của bộ tạo số rõ ràng phụ thuộc vào tính chính xác của hoành đồ cơ bản. Tính hính xác của hoành đồ (chi tiết chương sau) phụ thuộc vào số các mẫu khả dụng.

.4.3. Phương pháp loại bỏ

Kỹ thuật loại bỏ (hay tiếp nhận) để tạo các biến ngẫu nhiên có pdf mong muốn $f_X(x)$ hay (x) "*đích*", về cơ bản bao gồm việc lấy giới hạn pdf đích bởi hàm $Mg_X(x)$, trong đó $g_X(x)$ thể iện pdf của biến ngẫu nhiên được tạo ra một cách dễ dàng và M là hằng số đủ lớn để đảm bảo:

$$Mg_X(x) \ge f_X(x), \quad \forall x \tag{7.53}$$

Ở dạng đơn giản nhất $g_X(x)$ phân bố đều trên $(0,a)$. Nếu pdf đích $f_X(x) = 0$ ở ngoài khoảng $),a)$, thì ta có:

$$Mg_X(x) = \begin{cases} b = \dfrac{M}{a}, & 0 \le x \le a \\ 0, & \text{nếu khác} \end{cases} \tag{7.54}$$

Trong đó, do $Mg(x)$ giới hạn $f_X(x)$:

$$b = \frac{M}{a} \ge \max\{f_X(x)\} \tag{7.55}$$

Điều này được minh họa ở hình 7.10.

Thuật toán tạo biến ngẫu nhiên X có pdf là $f_X(x)$ được xác định bởi 4 bước sau:

1. Tạo U_1 và U_2 phân bố đều trong khoảng (0,1)

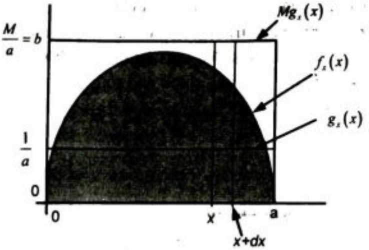
2. Tạo V_1 phân bố đều trong khoảng $(0,a)$, trong đó a là giá trị lớn nhất của X

3. Tạo V_2 phân bố đều trong khoảng $(0,b)$, trong đó b ít nhất là giá trị lớn nhất của $f_X(x)$

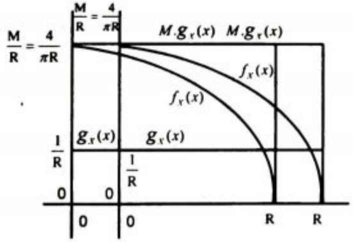
4. Nếu $V_2 \le f_X(V_1)$, thì đặt $X = V_1$. Nếu không thỏa mãn bất đẳng thức thì hủy bỏ V_1 và V_2 quá trình được lặp lại từ bước 1.

Dễ dàng chỉ ra rằng thủ tục này tạo ra biến ngẫu nhiên X có pdf đích là $f_X(x)$.

Vì V_1 và V_2 được phân bố đều, nên điểm được tạo ra bởi cặp mẫu (V_1, V_2) có xác suất rơi o mọi nơi trong diện tích ab là như nhau. Xác suất mà V_1 được chấp nhận là phần diện tích ab m dưới pdf $f_X(x)$. Đây là tỉ số giữa diện tích có nền xám trong hình 7.10 với toàn bộ diện tích . Vì vậy:



Hình 7.10: Phương pháp loại bỏ



Hình 7.11: Phương pháp loại bỏ cho ví dụ 7.9

$$\Pr\{V_1 \text{ được chấp nhận}\} = \frac{\int_0^a f_X(x)dx}{ab} \tag{7.56}$$

Theo định nghĩa, tử số trong biểu thức trên bằng 1, nên:

$$\Pr\{V_1 \text{ được chấp nhận}\} = \frac{1}{ab} = \frac{1}{M} \tag{7.57}$$

Vì vậy:

$$P\{x < V_1 \le x + dx | V_1 \text{ được chấp nhận}\} = \frac{\Pr\{x < V_1 \le x + dx \text{ , } V_1 \text{ được chấp nhận}\}}{\Pr\{V_1 \text{được chấp nhận}\}}$$

$$= \frac{f_X(x)dx / ab}{1/ab} \tag{7.58}$$

$$= f_X(x)dx = \Pr\{x < X \le x + dx\} \tag{7.59}$$

định nghĩa pdf đích.

Ví dụ 7.9: Trong ví dụ này ta áp dụng kỹ thuật đã được đề cập cho pdf sau

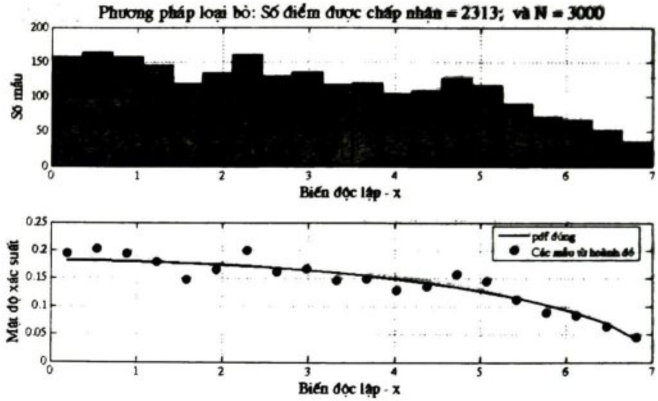
$$f_X(x) = \begin{cases} \frac{4}{\pi R^2}\sqrt{R^2 - x^2}, & 0 \le x \le R \\ 0, & \text{nếu khác} \end{cases} \tag{7.60}$$

Ta tìm thuật toán để tạo biến ngẫu nhiên X có pdf $f_X(x)$ bằng cách đặt $a = R$ và $b = M/R$. Hình 7.10 được sửa đổi cho trường hợp cụ thể này được minh hoạ trong hình 7.11.

Mã chương trình Matlab được cho bởi file **NVD7_rejex1.m** (có trong Phụ lục 7A). Kết quả chạy chương trình với N = 3000 điểm được cho ở hình 7.12. Trong 3000 điểm thì 2301 điểm được chấp nhận và 699 điểm bị loại bỏ, hiệu quả 76,70%, nó gần với giá trị lý thuyết (khi $N \to \infty$) là 78,54%.

Mặc dù ta đã minh hoạ phương pháp loại bỏ cho trường hợp trong đó pdf lấy giới hạn là đều, nhưng kỹ thuật được minh hoạ ở đây dễ dàng sửa đổi cho trường hợp trong đó pdf lấy giới

hạn là không đều. Dưới dạng lý tưởng, pdf đích nên được lấy giới hạn chặt sao cho xác suất loại bỏ được giảm thiểu. Phương pháp loại bỏ làm việc tốt trong các trường hợp trong đó pdf đích có sự hỗ trợ hữu hạn (chỉ khác không trên dải hữu hạn). Tuy nhiên, sự hỗ trợ hữu hạn là không cần thiết và kỹ thuật loại bỏ, như được minh hoạ ở đây, có thể dễ dàng được mở rộng cho trường hợp hỗ trợ vô hạn.



Hình 7.12: Kết quả mô phỏng minh họa phương pháp loại bỏ

7.5. Tạo số ngẫu nhiên Gausơ không tương quan

Thấy rõ từ việc nghiên nghiên cứu các hệ thống truyền thông, thường xuyên gặp phải biến ngẫu nhiên Gausơ và tỏ ra là mô hình thích hợp cho tạp âm nhiệt và nhiều hiện tượng khác. Bộ tạo tạp âm Gausơ là khối cơ bản trong nhiều mô phỏng, kết quả là nhiều kỹ thuật đã được triển khai để tạo biến ngẫu nhiên Gausơ. CDF là:

$$F_X(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} dy = 1 - Q\left(\frac{x}{\sigma}\right) \tag{7.61}$$

Trong đó Q(.) là hàm Q Gausơ được định nghĩa là:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{-\frac{y^2}{2}} dy \tag{7.62}$$

Vì không thể viết hàm Q Gausơ ở dạng kín, nên không dùng được kỹ thuật biến đổi ngược. Có thể áp dụng kỹ thuật loại bỏ nhưng không hiệu quả. Vì vậy, cần phải tìm kỹ thuật khác để tạo biến ngẫu nhiên Gausơ.

7.5.1. Phương pháp lấy tổng các biến ngẫu nhiên phân bố đều

Lý thuyết giới hạn trung tâm (CLT) cho ta một phương pháp tốt để triển khai biến ngẫu nhiên có pdf phân bố Gausơ. CLT phát biểu rằng, dưới các điều kiện khá tổng quát thì pdf của tổng N biến ngẫu nhiên độc lập sẽ hội tụ về biến ngẫu nhiên Gausơ khi $N \to \infty$.

Giả sử có N biến ngẫu nhiên đều độc lập U_i, $i = 1, 2, ... N$. Từ N biến ngẫu nhiên đều này tạo thành:

$$Y = B\sum_{i=0}^{N}\left(U_i - \tfrac{1}{2}\right) \tag{7.63}$$

Trong đó B là hằng số thiết lập phương sai của Y. Từ CLT ta biết rằng Y hội tụ về biến ngẫu nhiên Gausơ khi $N \to \infty$. Vì $E\{U_i\} = \frac{1}{2}$ nên trung bình của Y là:

$$E\{Y\} = B\sum_{i=0}^{N}\left(E\{U_i\} - \frac{1}{2}\right) = 0 \tag{7.64}$$

Tìm được phương sai của Y bằng cách lưu ý đến phương sai của U_i-$1/2$ là:

$$\text{var}\left\{U_i - \frac{1}{2}\right\} = \int_{-1/2}^{1/2} x^2 dx = \frac{1}{12} \tag{7.65}$$

Vì các biến ngẫu nhiên thành phần U_i được giả định là độc lập nên:

$$\sigma_y^2 = B^2\sum_{i=1}^{N}\text{var}\left\{U_i - \frac{1}{2}\right\} \tag{7.66}$$

Ta có:

$$\sigma_y^2 = \frac{NB^2}{12} \tag{7.67}$$

Vì vậy khi cho trước giá trị của N, thì phương sai của Y là σ_y^2 có thể được đặt một giá trị mong muốn bất kỳ bằng cách chọn B phù hợp. Việc chọn N là sự dung hoà giữa tốc độ và độ chính xác của các đuôi của pdf kết quả. Giá trị N thường được thiết lập là 12 vì $N = 12$ cho kết quả đơn giản $B = \sigma_y$.

Trong khi thủ tục tạo ra biến ngẫu nhiên Gausơ dựa vào định lý giới hạn tập trung là dễ hiểu, thì xuất hiện một số khó khi áp dụng vào các bài toán truyền tham số thực tế đó là:

Thứ nhất: vì $\left(U_i - \frac{1}{2}\right)$ thay đổi từ $-\frac{1}{2}$ đến $\frac{1}{2}$, nên theo (7.63) Y thay đổi từ $-\frac{BN}{2}$ đến $\frac{BN}{2}$. Theo đó, mặc dù (7.63) có thể xấp xỉ khá chính xác đến pdf của biến ngẫu nhiên Gausơ ở lân cận giá trị trung bình, thì các đuôi của pdf bị cắt bỏ ở bên ngoài $\pm\frac{BN}{2}$. Nếu mục đích mô phỏng hiệu năng xác suất lỗi ký hiệu của hệ thống truyền thông, thì các đuôi của pdf là rất quan trọng (vì các đuôi của pdf thể hiện các giá trị tạp âm gây ra các lỗi truyền dẫn). Có thể giảm thiểu ảnh hưởng của việc cắt các đuôi pdf của Y bằng cách chọn N đủ lớn.

Cụ thể, từ (7.67) giá trị của B là:

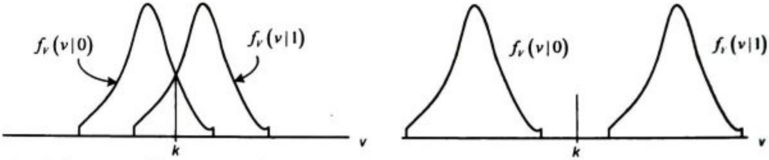
$$B = \sigma_y\sqrt{\frac{12}{N}} \tag{7.68}$$

Vì vậy, pdf "xấp xỉ Gausơ" được cắt sao cho nó chỉ khác 0 trong khoảng:

$$\pm\sigma_y \frac{N}{2}\sqrt{\frac{12}{N}} = \pm\sigma_y\sqrt{3N} \tag{7.69}$$

Với $N = 100$ thì biến ngẫu nhiên Gausơ bị cắt tại $\pm 17{,}32\sigma_y$. Đối với một số ứng dụng việc cắt tại 17 lần độ lệch chuẩn có thể vẫn mang lại các lỗi đáng kể. Giá trị phù hợp của N là *phụ thuộc ứng dụng*.

Khó khăn của việc cắt các đuôi pdf của tạp âm trong hệ thống truyền thông số được minh hoạ trong hình 7.13, cho thấy các pdf có điều kiện tại đầu ra của máy thu lọc thích hợp khi SNR đầu vào máy thu nhỏ và khi SNR đầu vào máy thu lớn (các đuôi của các pdf thực tế liên tục, hình 7.13 nhấn mạnh ảnh hưởng của cắt xén). Các pdf với điều kiện bit 0 nhị phân được phát đi ký hiệu là $f_V(v/0)$, và với điều kiện bit 1 nhị phân được phát đi ký hiệu là $f_V(v/1)$. Hình 7.13 được xây dựng với giả định phương sai tạp âm là không đổi và SNR được điều chỉnh bằng cách thay đổi công suất tín hiệu. Trường hợp SNR đầu vào máy thu đủ nhỏ như hình 7.13(a) thì các pdf điều kiện chồng lấn nhau đáng kể và xác suất lỗi có thể được xác định với độ chính xác hợp lý. Khi công suất tín hiệu tăng, thì đẩy các pdf điều kiện xa nhau và độ chính xác mô phỏng giảm đi. Do việc cắt các đuôi của các pdf có điều kiện, tăng công suất tín hiệu dẫn đến tình huống các pdf có điều kiện không còn chồng lấn nhau nữa, được minh hoạ ở hình 7.13(b). Nếu các pdf có điều kiện không chồng lấn nhau thì xác suất lỗi bằng 0 độc lập với SNR, rõ ràng là phi thực tế.



(a) pdf đầu ra bộ lọc phối hợp khi SNR đầu vào máy thu nhỏ (b) pdf đầu ra bộ lọc phối hợp khi SNR đầu vào máy thu lớn

Hình 7.13: Ảnh hưởng của việc cắt xén hàm mật độ xác suất pdf đầu ra bộ lọc phối hợp khi SNR đầu vào máy thu thấp và cao

Thứ hai: vì chọn N lớn khi sử dụng (7.63) để xấp xỉ biến ngẫu nhiên Gausơ. Vì nó thực hiện N lần gọi bộ tạo số ngẫu nhiên phân bố đều để tạo một giá trị của Y, nên khi dùng thuật toán (7.63) yêu cầu thời gian CPU vượt quá.

Hai đặc tính bù lại của (7.63): *(i)* nó làm tốt việc xấp xỉ biến ngẫu nhiên Gausơ ở lân cận trung bình của Y; *(ii)* Y sẽ xấp xỉ Gausơ cả khi pdf của các biến ngẫu nhiên U_i thành phần là phân bố không đều.

Trong khi ta chủ yếu đề cập trong mô phỏng dùng xử lý nối tiếp bằng một CPU duy nhất, thì điều này cho thấy các thuật toán không phù hợp với các ứng dụng xử lý nối tiếp truyền thống nhưng khá phù hợp với các cơ chế xử lý song song. Ví dụ nếu cơ chế xử lý song song nào đó sử dụng 100 CPU, thì nó có thể tạo 100 giá trị của một biến ngẫu nhiên đều tại cùng một thời điểm. Tổng 100 biến ngẫu nhiên đều dẫn đến xấp xỉ hoàn hảo cho biến ngẫu nhiên Gausơ

đối với hầu hết các ứng dụng và xử lý song song được thực hiện nhanh chóng. Đây chỉ là một minh chứng cho việc lựa chọn thuật toán phụ thuộc vào môi trường tính toán.

7.5.2. Ánh xạ biến ngẫu nhiên Rayleigh thành biến ngẫu nhiên Gausơ

Từ ví dụ 7.8 cho thấy, biến ngẫu nhiên Rayleigh R có thể được tạo ra từ biến ngẫu nhiên đều U bằng quan hệ $R = \sqrt{-2\sigma^2 \ln U}$. Tại đây, ta xét bài toán ánh xạ biến ngẫu nhiên Rayleigh sang biến ngẫu nhiên Gausơ.

Giả sử X và Y là 2 biến ngẫu nhiên Gausơ độc lập có cùng phương sai σ^2. Vì X và Y là độc lập nên pdf đồng thời là tích của các pdf duyên biên. Vì vậy:

$$\begin{aligned} f_{XY}(x,y) &= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} \\ &= \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \end{aligned} \tag{7.70}$$

Với x = rcosθ và y = rsinθ ta có:

$$x^2 + y^2 = r^2 \tag{7.71}$$

$$\theta = \tan^{-1}\left(\frac{y}{x}\right) \tag{7.72}$$

pdf đồng thời $f_{R\Theta}(r,\theta)$ tìm được từ $f_{XY}(x,y)$ bởi phép biến đổi:

$$f_{R\Theta}(r,\theta) dA_{R\Theta} = f_{XY}(x,y) dA_{XY} \tag{7.73}$$

Trong đó $dA_{R\Theta}$ và dA_{XY} biểu diễn các diện tích vi phân trong các mặt phẳng R, Θ và X,Y. Theo (7.73) ta có:

$$f_{R\Theta}(r,\theta) = f_{XY}(x,y) \frac{dA_{XY}}{dA_{R\Theta}}\bigg|_{\substack{x=r\cos\theta \\ y=r\sin\theta}} \tag{7.74}$$

Tỉ số của các diện tích vi phân là Jacobian của biến đổi, nó là:

$$\frac{dA_{XY}}{dA_{R\Theta}} = \frac{\partial(x,y)}{\partial(r,\theta)} = \begin{vmatrix} dx/dr & dx/d\theta \\ dy/dr & dy/d\theta \end{vmatrix} \tag{7.75}$$

Cho ta:

$$\frac{dA_{XY}}{dA_{R\Theta}} = \begin{vmatrix} \cos\theta & -r\sin\theta \\ \sin\theta & r\cos\theta \end{vmatrix} = r \tag{7.76}$$

Vì vậy thay (7.70), (7.71) và (7.76) vào (7.74) ta được kết quả:

$$f_{R\Theta}(r,\theta) = \frac{r}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}}, \quad 0 \le r < \infty, \quad 0 \le \theta < 2\pi \tag{7.77}$$

Bây giờ ta khảo sát các pdf duyên biên của R và Θ :

pdf của R là:

$$f_R(r) = \int_0^{2\pi} \frac{r}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}} d\theta = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}}, \qquad 0 \le r < \infty \tag{7.78}$$

pdf của Θ là:

$$f_\Theta(\theta) = \int_0^{\infty} \frac{r}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}} dr = \frac{1}{2\pi}, \qquad 0 \le \theta < 2\pi \tag{7.79}$$

Do đó R là biến ngẫu nhiên phân bố Rayleigh và Θ là biến ngẫu nhiên phân bố đều. Vì biến ngẫu nhiên Rayleigh được tạo ra từ 2 biến ngẫu nhiên Gausơ *trực giao*, theo đó các phép chiếu trực giao của biến ngẫu nhiên Rayleigh tạo ra cặp biến ngẫu nhiên Gausơ. Vì vậy, giả sử R là biến ngẫu nhiên Rayleigh và Θ là biến ngẫu nhiên đều trên khoảng $(0,2\pi)$, thì các biến ngẫu nhiên Gausơ X và Y được tạo ra bằng các quan hệ sau:

$$X = R\cos\Theta \tag{7.80}$$

$$Y = R\sin\Theta \tag{7.81}$$

Cả X và Y đều là các biến ngẫu nhiên Gausơ trung bình 0 và có cùng phương sai σ^2. Vì chúng là các biến ngẫu nhiên Gausơ và không tương quan, nên X và Y *độc lập thống kê* (lưu ý rằng, tính chất của biến ngẫu nhiên Gausơ là không tương quan thì độc lập thống kê, tính chất này không có được đối với các phân bố khác). Vì vậy, cặp biến ngẫu nhiên Gausơ độc lập X và Y được tạo ra từ cặp biến ngẫu nhiên phân bố đều U_1 và U_2 bằng thuật toán:

$$X = \sqrt{-2\sigma^2 \ln(U_1)}.\cos(2\pi U_2) \tag{7.82}$$

$$Y = \sqrt{-2\sigma^2 \ln(U_1)}.\sin(2\pi U_2) \tag{7.83}$$

trong đó ta đã sử dụng (7.46) cho R.

Chương trình Matlab thực hiện thuật toán Box-Muller được cho ở file **NVD7_boxmu1.m** (có trong Phụ lục 7A).

7.5.3. Phương pháp cực

Thuật toán khác để tạo cặp biến ngẫu nhiên Gausơ không tương quan trung bình 0 là phương pháp cực. Thuật toán bao gồm các bước sau:

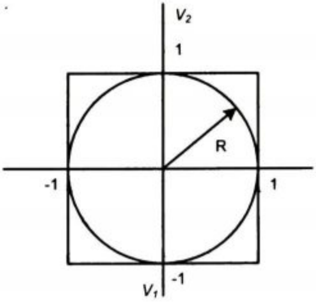
1. Tạo 2 biến ngẫu nhiên độc lập U_1 và U_2, cả hai đều được phân bố đều trong khoảng (0,1).
2. Đặt $V_1 = 2U_1 - 1$ và $V_2 = 2U_2 - 1$ sao cho V_1 và V_2 là độc lập thống kê nhau và được phân bố đều trong khoảng (-1,1).
3. Tạo $S = \sqrt{V_1^2 + V_2^2}$. Nếu $S<1$ tiếp tục thực hiện bước 4. Nếu $S \ge 1$ huỷ S và quay trở lại bước 1.

4. Tạo $A(S)=\sqrt{\left(-2\sigma^2 \ln S\right)/S}$.

5. Đặt $X = A(S)V_1$ và $Y = A(S)V_2$.

Mã chương trình Matlab tạo cặp véc-tơ ngẫu nhiên Gausơ sử dụng phương pháp cực được cho bởi file **NVD7_ polar.m** (có trong Phụ lục 7A).

Lưu ý rằng, ta đã dùng hàm **rand** có sẵn trong thư viện Matlab để tạo cặp biến ngẫu nhiên đều trong chương trình **NVD7_ polar.m**.



Hình 7.14: Phương pháp cực

Phương pháp cực là một ví dụ của phương pháp loại bỏ, vì từ bước 3 một số giá trị của S bị loại bỏ. Do đó, sẽ tạo ra ít hơn N biến ngẫu nhiên mỗi khi gọi hàm này. Dễ dàng xác định được xác suất loại bỏ từ hình 7.14. Các biến ngẫu nhiên V_1 và V_2 được phân bố đều trong hộp có diện tích $A_{hộp} = 4$ giới hạn đường tròn bán kính $R = 1$ và diện tích $A_{vòng\ tròn} = \pi$. Vì vậy, xác suất mà S bị loại bỏ là:

$$\Pr\{\text{loại bỏ}\} = 1-\frac{A_{\text{vòng tròn}}}{A_{\text{hộp}}} = 0{,}2146 \tag{7.84}$$

Thuật toán cực thường cung cấp các kết quả có các tính chất tương quan tốt hơn thuật toán Box-Muller. Tuy nhiên, có một nhược điểm của phương pháp cực. Lưu ý rằng, N lần gọi đối với thuật toán Box-Muller sẽ tạo N cặp biến ngẫu nhiên Gausơ. Nếu thuật toán cực được gọi N lần, thì số cặp biến ngẫu nhiên Gausơ được tạo ra sẽ là *một biến ngẫu nhiên* có trung bình $(\pi/4)N$. Thực tế phải thực hiện nhiều lần gọi không được biết trước cho thuật toán cực để tạo một số biến ngẫu nhiên Gausơ cho trước thường làm phức tạp chương trình mô phỏng.

7.5.4. Thực hiện Matlab

Trước phiên bản Matlab 5, con số ngẫu nhiên Gausơ được chứa trong thư viện Matlab, **randn,** sử dụng bộ tạo số tiêu chuẩn nhỏ nhất theo (7.24) và dùng phương pháp cực để ánh xạ các số ngẫu nhiên phân bố đều sang các số ngẫu nhiên có phân bố Gausơ. Bắt đầu từ Matlab 5, một thuật toán hoàn toàn khác không chứa các phép nhân hoặc phép chia. Bộ tạo số ngẫu nhiên mới trực tiếp tạo ra các số có pdf phân bố Gausơ không cần phải chuyển đổi các số ngẫu nhiên phân bố đều. Vì không dùng phép nhân và phép chia trong thuật toán này, và không cần bước chuyển từ phân bố đều sang phân bố Gausơ nên rất nhanh. Thuật toán được dùng trong các phiên bản sau của Matlab là một phiên bản cải tiến của thuật toán Ziggurt cơ bản.

7.6. Tạo số ngẫu nhiên Gausơ tương quan

Ta mới chỉ tạo ra các số ngẫu nhiên *không tương quan*. Dưới đây, ta xét việc tạo các số ngẫu nhiên có pdf phân bố Gausơ và *tương quan*. *Trước hết*, ta nghiên cứu kỹ thuật đơn giản tạo 2 chuỗi liên quan với nhau thông qua hệ số tương quan cho trước. *Sau đó*, ta xét trường hợp tổng quát hơn ở đó chuỗi được tạo ra có PSD cho trước. Tất nhiên, việc thiết lập PSD cho trước là tương đương với việc thiết lập hàm tự tương quan cho trước.

7.6.1. Thiết lập hệ số tương quan

Ta đã thấy 2 phương pháp tạo cặp biến ngẫu nhiên Gausơ không tương quan. Dễ dàng ánh xạ cặp biến ngẫu nhiên Gausơ không tương quan X và Y thành cặp biến ngẫu nhiên Gausơ có mức độ tương quan cụ thể. Giả sử X và Y có trung bình 0 và không tương quan thì bước tiếp theo là tạo biến ngẫu nhiên thứ 3 được định nghĩa bởi:

$$Z = \rho X + \sqrt{1-\rho^2}Y \tag{7.85}$$

Trong đó ρ là tham số với $|\rho| \le 1$. Với thuật toán này, X và Y là các biến ngẫu nhiên trung bình 0 có phương sai bằng nhau. Ta sẽ chỉ ra rằng ρ là hệ số tương quan liên hệ X và Z.

Việc chứng minh là đơn giản. Thấy rõ Z là biến ngẫu nhiên Gausơ bởi lẽ nó là kết hợp tuyến tính của các biến ngẫu nhiên Gausơ. Theo đó, Z cũng có trung bình 0 nếu X và Y có trung bình 0. Phương sai của Z là:

$$\begin{aligned}\sigma_Z^2 &= E\left\{\left[\rho X + \sqrt{1-\rho^2}Y\right]^2\right\} \\ &= \rho^2.E\{X^2\} + 2\rho\sqrt{1-\rho^2}.E\{XY\} + (1-\rho^2).E\{Y^2\}\end{aligned} \tag{7.86}$$

Vì $E\{XY\} = E\{Y\}.E\{Y\} = 0$ và $\sigma_X^2 = \sigma_Y^2 = \sigma^2$ nên ta có:

$$\sigma_Z^2 = \rho^2\sigma^2 + (1-\rho^2)\sigma^2 = \sigma^2 \tag{7.87}$$

Hiệp biến (đồng phương sai) E{XZ} là:

$$\begin{aligned}E\{XZ\} &= E\{X[\rho X + (1-\rho)Y]\} = \rho E\{X^2\} + (1-\rho)E\{XY\} \\ &= \rho E(X^2) = \rho\sigma^2\end{aligned} \tag{7.88}$$

Trong đó lưu ý rằng, *X* và *Y* là độc lập và có trung bình 0. Hệ số tương quan ρ_{XZ} là:

$$\rho_{XZ} = \frac{E\{XZ\}}{\sigma_X\sigma_Z} = \frac{\rho\sigma^2}{\sigma^2} = \rho \tag{7.89}$$

Như được mong đợi.

7.6.2. Thiết lập hàm tự tương quan hoặc mật độ phổ công suất tuỳ ý

Kỹ thuật tổng quát để thiết lập chuỗi các số ngẫu nhiên có hàm tự tương quan cho trước, hoặc tương đương với PSD cho trước là *lọc* tập các mẫu không tương quan sao cho thiết lập được PSD đích. Theo định nghĩa, các mẫu không tương quan có PSD là không đổi (hằng số) trong độ rộng băng tần mô phỏng $|f|<f_s/2$. Theo định nghĩa, phương sai là diện tích nằm dưới hàm $S_n(f)$ được minh hoạ trong hình 7.15:

$$\sigma_n^2 = \frac{1}{2}N_0 f_s \tag{7.90}$$

Vì vậy, để thiết lập PSD tạp âm cho trước N_0, thì phương sai bộ tạo số ngẫu nhiên (tạp âm) và tần số lấy mẫu phải thoả mãn:

$$f_s = \frac{2\sigma_n^2}{N_0} \tag{7.91}$$

Vì vậy, phương sai bộ tạo tạp âm khi cho trước PSD là hàm của tần số lấy mẫu.

Việc thiết lập PSD mong muốn cho chuỗi ngẫu nhiên là tương đối dễ hiểu bằng cách sử dụng kết quả cơ bản từ lý thuyết quá trình ngẫu nhiên cơ bản. Ta biết rằng, nếu đưa vào hệ thống tuyến tính một quá trình ngẫu nhiên có PSD là $S_X(f)$, thì PSD đầu ra hệ thống là:

$$S_Y(f) = |H(f)|^2 S_X(f) \tag{7.92}$$

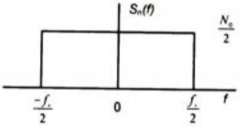
Trong đó *H(f)* là hàm truyền đạt của hệ thống tuyến tính. Điều này được minh hoạ ở hình 7.16. Nếu các mẫu tạp âm là độc lập thì PSD đầu vào là không đổi tại $K = N_0/2$ *[W/Hz]*:

$$S_Y(f) = |H(f)|^2 K \tag{7.93}$$

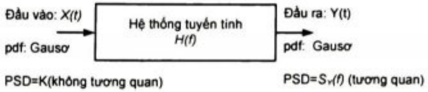
Và cần có *H(f)* để thiết lập PSD đích là:

$$H(f) = \sqrt{\frac{S_Y(f)}{K}} \tag{7.94}$$

Vì vậy, bài toán định dạng mật độ phổ công suất nhằm đáp ứng yêu cầu cho trước quy về bài toán tìm bộ lọc có hàm truyền đạt *H(f)* sao cho $H^2(f)$ tạo ra định dạng phổ theo yêu cầu.



Hình 7.15: PSD đối với các mẫu độc lập



Hình 7.16: Tạo chuỗi ngẫu nhiên tương quan

Ví dụ 7.10: Phương pháp thông dụng để tổng hợp một bộ lọc có hàm truyền đạt cho trước là xác định sự phù hợp nhất với hàm truyền đạt sao cho lỗi bình phương trung bình cực tiểu MMSE. Giải các phương trình Yule-Walker bằng hàm Matlab *yulewalk* cho vấn đề này. Cụ thể, *yulewalk* xác định hệ số lọc b_k và a_k sao cho hàm truyền đạt:

$$H(z) = \frac{b_0 + b_1 z^{-1} + \cdots + b_n z^{-n}}{1 + a_1 z^{-1} + \cdots + a_n z^{-n}} \tag{7.95}$$

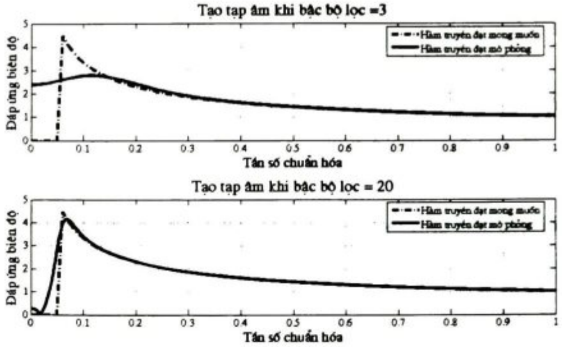
Có lỗi bình phương trung bình cực tiểu MMSE (lưu ý rằng, MSE là lấy trung bình của lỗi được bình phương) phù hợp với hàm truyền đạt cho trước.

Để minh hoạ kỹ thuật này, giả sử cho trước PSD có dạng $S(f) = K/f$, thường được dùng để mô hình hóa tạp âm pha trong các bộ tạo dao động. Để tạo tạp âm này, ta phải tạo một bộ lọc có hàm truyền đạt dạng $H(f) = K/\sqrt{f}$. Cho tạp âm trắng qua bộ lọc này sẽ tạo ra quá trình tạp âm theo yêu cầu. Vì $H(f) \to \infty$ khi $f \to 0$, nên hàm truyền đạt được định nghĩa là:

$$H(f)=\begin{cases}0, & |f|<f_0\\ \dfrac{K}{\sqrt{f}}, & |f|>f_0\end{cases} \tag{7.96}$$

Mã chương trình Matlab **NVD7_ flicker.m** (có trong Phụ lục 7A) tạo ra một xấp xỉ cho *H(f)* trong đó tần số được chuẩn hoá theo tần số Nyquist f_N và $f_0 = f_N/20$:

Kết quả chạy chương trình **NVD7_ flicker.m** được minh hoạ ở hình 7.17, trong đó hàm truyền đạt mong muốn được minh hoạ bằng đường nét đứt. Xấp xỉ bậc 3 được minh hoạ ở phần trên và xấp xỉ bậc 20 được minh hoạ ở phần dưới. Trong vùng tần số ở đó H(f) thay đổi nhanh thì cần có xấp xỉ bậc cao hơn. Giải pháp này là công cụ mạnh để tạo các PSD tùy ý.



Hình 7.17: Tạo tạp âm

Ví dụ 7.11: Mô phỏng các hệ thống vô tuyến trong đó có tính di động, cần phải tạo một quá trình có mật độ phổ công suất PSD:

$$S(f)=\begin{cases}\dfrac{1}{\sqrt{1-(f/f_d)^2}}, & |f|<f_d\\ 0, & \text{nếu khác}\end{cases} \tag{7.97}$$

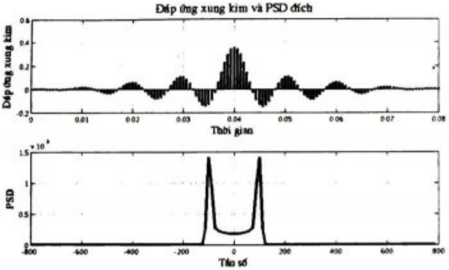
Để thể hiện hiệu ứng dopper. Đại lượng f_d trong (7.97) thể hiện tần số Doppler lớn nhất. Như được minh hoạ trong (7.94), hàm truyền đạt bộ lọc là:

$$H(f)=\begin{cases}\left[1-(f/f_d)^2\right]^{-1/4}, & |f|<f_d\\ 0, & \text{nếu khác}\end{cases} \tag{7.98}$$

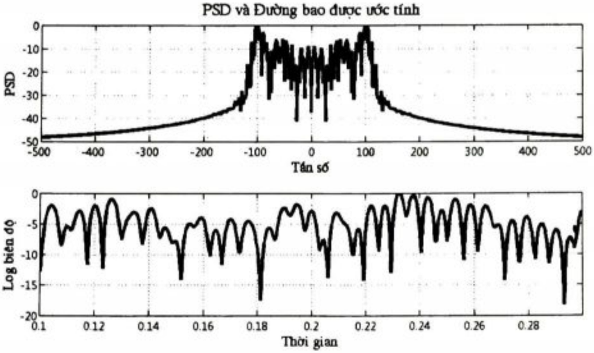
Bộ lọc này được thực hiện như bộ lọc FIR có đáp ứng xung đạt được bằng cách thực hiện IDFT lên các giá trị mẫu của (7.98).

Mã chương trình Matlab tạo đáp ứng xung kim của bộ lọc Jakes, và việc lọc tạp âm trắng bằng bộ lọc, được cho bởi file **NVD7_Jakes.m** có trong Phụ lục 7A. Kết quả chạy chương trình

NVD7_Jakes.m được cho ở các hình 7.18 và 7.19. Hình 7.18 minh hoạ đáp ứng xung và hàm truyền đạt H(f). Hình 7.19 minh hoạ ảnh hưởng của việc cho tạp âm trắng phức qua bộ lọc. Phần trên chỉ ra rằng PSD được ước tính tại đầu ra bộ lọc. Phần dưới minh hoạ độ lớn của đường bao trên thang loga. Trong hệ thống truyền thông vô tuyến, tương ứng với đường bao pha đinh.



Hình 7.18: Đáp ứng xung kim và PSD đích

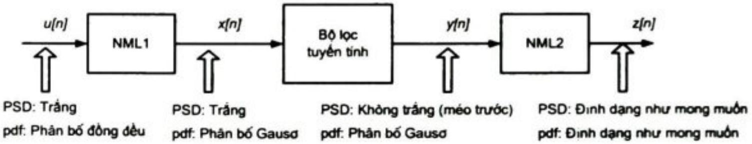


Hình 7.19: PSD được ước tính và hàm đường bao

7.7. Thiết lập hàm mật độ xác suất và mật độ phổ công suất

Ở dạng tổng quát, tạo dạng sóng tạp âm cần phải đáp ứng đồng thời các yêu cầu về pdf và PSD là một nhiệm vụ khó khăn. Tuy nhiên bài toán sẽ đơn giản, nếu pdf đầu ra hệ thống là Gausơ thì chỉ cần tạo một chuỗi mẫu cho đầu vào bộ lọc trong đó các mẫu là Gausơ và độc lập. Vì các mẫu đầu vào là độc lập nên PSD sẽ là hằng số tại K [W/Hz]. PSD có thể được định dạng bằng việc dùng bộ lọc tuyến tính như đã mô tả. Vì bộ lọc định dạng mật độ phổ công suất PSD là tuyến tính nên hàm mật độ xác suất pdf của đầu ra sẽ là Gausơ. Có được kết quả đơn giản bởi lẽ *bất kỳ biến đổi tuyến tính của một quá trình Gausơ sẽ tạo ra một quá trình Gausơ khác*. Tuy nhiên, nhiều bài toán thực tế có thể giải quyết được theo cách này.

Nếu cả pdf và PSD của dạng sóng đích được định rõ và yêu cầu pdf khác với Gausơ thì bài toán khó hơn rất nhiều. Lời giải cho bài toán này được đề xuất bởi Sondhi[12]. Sơ đồ cơ bản để thực hiện thuật toán Sondhi được minh hoạ ở hình 7.20. Như thường lệ, ta bắt đầu bằng một chuỗi các mẫu *u[n]* phân bố đều trong khoảng (0,1). Ngoài ra, chuỗi {*u[n]*} được coi là tương quan delta sao cho PSD là trắng. Nhiệm vụ của phần tử phi tuyến không nhớ đầu tiên MNL1 là ánh xạ chuỗi {*u[n]*} thành chuỗi {*x[n]*}({x[n]}là trắng nhưng có pdf phân bố Gausơơ. Ta đã nghiên cứu một số kỹ thuật để thực thi ánh xạ này. Hoạt động lọc là làm méo trước phổ sao cho PSD là $S_Y(f)$. Vì bộ lọc là tuyến tính, nên pdf của chuỗi {*y[n]*} vẫn là Gausơ. Hoạt động cơ bản của phần tử phi tuyến không nhớ thứ hai MNL2 là ánh xạ pdf phân bố Gausơ của chuỗi {*y[n]*} thành pdf mong muốn cuối cùng. Tuy nhiên, tính phi tuyến thứ hai cũng ảnh hưởng lên $S_Y(f)$. Vì vậy, bộ lọc phải sửa đổi PSD đã được làm méo trước $S_Y(f)$ sao cho khi cho chuỗi mẫu qua thiết bị phi tuyến thứ hai làm cho chuỗi {*z[n]*} có cả PSD và pdf mong muốn.

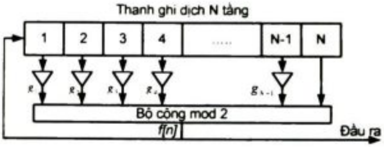


Hình 7.20: Thuật toán Sondhi

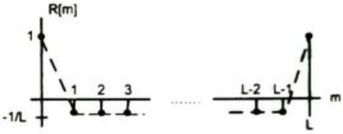
7.8. Các bộ tạo chuỗi PN

Các bộ tạo chuỗi giả ngẫu nhiên PN được dùng trong nhiều ứng dụng, đặc biệt trong lĩnh vực đồng bộ. Chẳng hạn, các chuỗi PN được dùng để xấp xỉ hóa biến ngẫu nhiên có hàm mật độ xác suất phân bố đều. Bộ tạo chuỗi PN có thể có một số dạng, nhưng dạng thông dụng nhất mà ta sẽ tập trung đề cập được minh hoạ trong hình 7.21.

Trong môi trường mô phỏng lý do quan trọng nhất để sử dụng chuỗi PN là mô hình hóa các nguồn dữ liệu. Bằng cách sử dụng bộ tạo chuỗi PN sẽ tạo ra hầu hết các kết hợp bit có chiều dài cho trước trên chiều dài mô phỏng ngắn nhất. Sẽ được dùng để nghiên cứu ảnh hưởng của giao thoa giữa các ký hiệu ISI trong chương 10 khi đó ta xét các kỹ thuật mô phỏng bán giải tích.



Hình 7.21: Bộ tạo chuỗi PN



Hình 7.22: Tự tương quan các nội dung thanh ghi dịch đối với chuỗi PN dài cực đại

Bộ tạo chuỗi PN gồm 3 phần cơ bản: thanh ghi dịch N tầng, bộ cộng mod-2, và véc tơ kết nối (xác định các kết nối giữa các tầng của thanh ghi dịch cụ thể với bộ cộng mod-2). Véc tơ kết nối thiết lập các đặc tính hiệu năng của bộ tạo chuỗi và được định nghĩa bởi đa thức:

$$g(D) = 1 + g_1 D + g_2 D^2 + ... + g_{N-1} D^{N-1} + D^N \quad (7.99)$$

Nếu $g_i = 1$ thì tầng thứ i của thanh ghi dịch được nối với bộ cộng mod-2. Nếu $g_i = 0$ thì không có kết nối. Lưu ý rằng, trong đa thức $g(D)$ cả g_0 và g_N đều bằng 1.

Có thể chỉ ra rằng chu kỳ lớn nhất của đầu ra bộ tạo chuỗi PN là:

$$L = 2^N - 1 \quad (7.100)$$

Và đạt được khi và chỉ khi đa thức $g(D)$ là đa thức nguyên thuỷ. Hàm tự tương quan $R[m]$ tại đầu ra bộ tạo chuỗi PN được minh hoạ trong hình 7.22, trong đó ta đã giả thiết các giá trị dữ liệu là ± 1. Khi $m = 0$ hoặc bội số của L thì hàm tự tương quan bằng 1. Khi $0 < m < L$ thì hàm tự tương quan $R[m] = -1/L$ và xấp xỉ bằng 0 khi L lớn. Theo đó, khi các chuỗi PN có chu kỳ *lớn*, thì hàm tự tương quan tiến dần tới ***xung kim***. Kết quả là, mật độ phổ công suất PSD xấp xỉ là trắng như mong muốn. Chuỗi PN có nhiều tính chất thú vị, một số tính chất quan tâm trong môi trường mô phỏng gồm:

- Chuỗi gần như được cân bằng. Nói cách khác, trong một chu kỳ của chuỗi thì số bit 1 sẽ lớn hơn số bit 0 là 1. Bằng cách thêm bit 0 bổ sung tại điểm bên phải thì chuỗi được cân bằng.

- Tất cả các kết hợp bit có thể có đều xuất hiện trong một chu kỳ, ngoại trừ sự kiện không có chuỗi N bit 0 nhưng có chuỗi N bit 1 (lưu ý rằng: nếu tất cả các bộ ghi dịch chứa số không nhị phân, thì bộ tạo sẽ trở nên bị kẹt trong trạng thái toàn không, bằng cách thêm số không ở điểm có N-1 số không, thì chuỗi trở nên cân bằng. Kết quả sẽ là chuỗi deBruijn).

- Hàm tự tương quan mặc dù tuần hoàn, nhưng gần giống như dạng sóng nhị phân ngẫu nhiên.

Việc thiết kế bộ tạo chuỗi PN dựa vào thanh ghi dịch N tầng quy về việc tìm đa thức nguyên thuỷ bậc N. Khi N lớn, đây có thể là một nhiệm vụ rất khó khăn. Tuy nhiên, các bảng mở rộng của các đa thức nguyên thuỷ có sẵn trong các tài liệu. Một số được biểu diễn phù hợp với (7.99) được cho ở bảng 7.1.

Bảng 7.1: Bảng một số đa thức nguyên thuỷ

N	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}	g_{11}	g_{12}	g_{13}	g_{14}
3	1	0	1											
4	1	0	0	1										
5	0	1	0	0	1									
6	1	0	0	0	0	1								
7	0	0	1	0	0	0	1							
8	0	1	1	1	0	0	0	1						
9	0	0	0	1	0	0	0	0	1					
10	0	0	1	0	0	0	0	0	0	1				
11	0	1	0	0	0	0	0	0	0	0	1			
12	1	0	0	1	0	1	0	0	0	0	0	1		
13	1	0	1	1	0	0	0	0	0	0	0	0	1	
14	1	0	0	0	0	1	0	0	0	1	0	0	0	1

Triển khai chương trình mô phỏng cho bộ tạo chuỗi PN là khá đơn giản. Trước hết, ta thể hiện nội dung thanh ghi dịch và véc-tơ kết nối bằng các véc-tơ sau:

$$B = \begin{bmatrix} b_1 & b_2 & .... & b_{N-1} & b_N \end{bmatrix} \quad (7.101)$$

$$G = \begin{bmatrix} g_1 & g_2 & .... & g_{N-1} & g_N \end{bmatrix} \quad (7.102)$$

Đầu ra bộ cộng mod-2 như được minh hoạ trong hình 7.21 là ký hiệu hồi tiếp *f[n]*, được định nghĩa bởi:

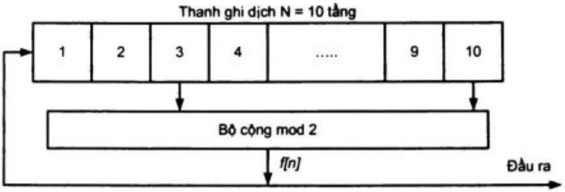
$$f[n] = \sum_{i=1}^{N} b_i g_i = B\left(G^T\right) \quad (7.103)$$

Tín hiệu hồi tiếp cũng là giá trị tiếp theo của b_1. Hiển nhiên, thanh ghi phải được khởi tạo bằng véc tơ B trong đó ít nhất phải có một giá trị $b_j \neq 0$.

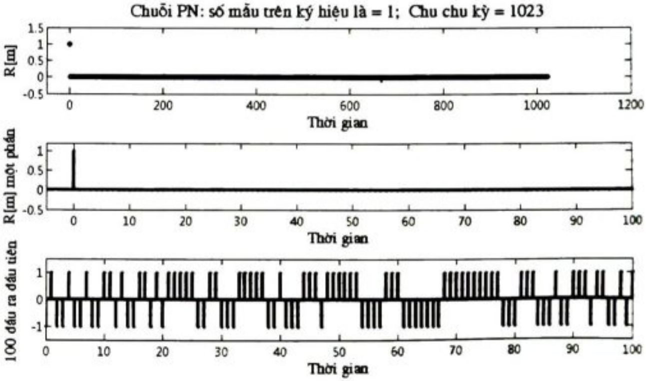
Ví dụ 7.12: Ta thiết kế bộ tạo chuỗi PN với $N = 10$. Với đa thức nguyên thủy *g(D)* và từ biểu thức (7.100), thì chu kỳ sẽ là $L = 1023$, từ bảng 7.1 véc tơ kết nối là:

$$G = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (7.104)$$

cho biết kết nối từ tầng thứ ba và tầng cuối cùng của thanh ghi dịch với bộ cộng mod-2, tạo ra cấu hình được cho ở hình 7.23. Mã chương trình Matlab mô phỏng bộ tạo chuỗi PN được cho ở file **NVD7_PNsim** (có trong Phụ lục 7A).

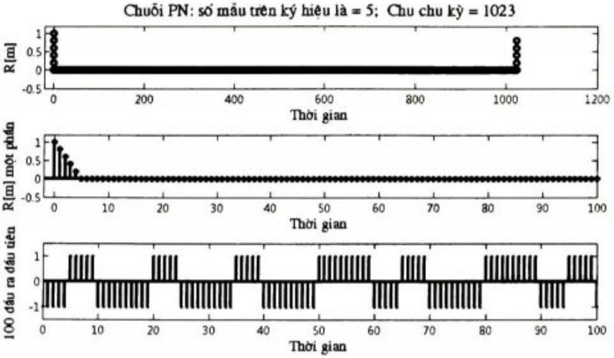


Hình 7.23: Bộ tạo chuỗi PN cho ví dụ 7.5



Hình 7.24: Biểu đồ được tạo khi lấy một mẫu trên một ký hiệu

Lưu ý rằng, chương trình thực hiện một kiểm tra để đảm bảo rằng bộ tạo chuỗi PN thực sự có chu kỳ đầy đủ (trường hợp này, chạy chương trình cho kết quả chu kỳ là 1023, là chu kỳ đầy đủ với $N = 10$). Chương trình cũng tạo ra đồ thị hàm tự tương quan, 101 mẫu đầu tiên của hàm tự tương quan, và 101 mẫu đầu tiên tại đầu ra bộ tạo. Khi lấy mẫu một mẫu/ký hiệu kết quả được minh hoạ trong hình 7.24. Thay đổi tốc độ lấy mẫu lên 5 mẫu cho kết quả trong hình 7.25.



Hình 7.25: Biểu đồ được tạo khi lấy năm mẫu trên một ký hiệu

7.9. Xử lý tín hiệu

Phần này ta nghiên cứu một số quan hệ vào/ra cho các hệ thống tuyến tính khi đầu vào/ra là ngẫu nhiên. Muốn vậy, cần quan tâm các kết quả cơ bản sau:

- Quan hệ giữa trung bình của đầu vào/ra hệ thống
- Quan hệ giữa phương sai của đầu vào/ra hệ thống
- Tương quan chéo vào/ra
- Quan hệ giữa tự tương quan và PSD của đầu vào hệ thống và đầu ra hệ thống.

Các quan hệ này rất hữu hiệu trong nghiên cứu mô phỏng, cho ta những công cụ cơ bản để xử lý tín hiệu cho trường hợp đầu vào hệ thống là một hàm mẫu của một quá trình ngẫu nhiên. Trong các phân tích dưới đây, ta coi rằng hệ thống là tuyến tính và cố định, và đầu vào hệ thống là quá trình dừng theo nghĩa rộng WSS. Bạn đọc nên tham khảo phần Phụ lục 7B1 và 7B2 có trong đĩa CD để được chi tiết hơn về xử lý các tín hiệu ngẫu nhiên.

7.9.1. Trung bình vào/ra

Vì hệ thống được giả định là tuyến tính, nên quan hệ các mẫu đầu ra $y[n]$ với các mẫu đầu vào $x[n]$ bởi tích chập rời rạc (tổng chập). Vì vậy, ta luôn có:

$$y[n] = \sum_{k=-\infty}^{\infty} h[k].x[n-k] \tag{7.105}$$

Theo định nghĩa, trung bình (giá trị một chiều DC) của đầu ra là:

$$E\{n[n]\} = E\left\{\sum_{k=-\infty}^{\infty} h[k].x[n-k]\right\} = \sum_{k=-\infty}^{\infty} h[k].E\{x[n-k]\} \qquad (7.106)$$

Vì kỳ vọng của một tổng là tổng các kỳ vọng, và cũng cần lưu ý rằng với hệ thống cố định thì các thành phần biểu diễn đáp ứng xung kim đơn vị *h[k]* là các hằng số. Từ giả định *dừng* ta có $E\{x[n-k]\} = E\{x[n]\}$. Dẫn đến kết quả đơn giản:

$$m_y = m_x \sum_{k=-\infty}^{\infty} h[k] \qquad (7.107)$$

Trong đó m_x và m_y là trung bình của *x[n]* và *y[n]*. Lưu ý rằng, vì $\sum_{k=-\infty}^{\infty} h[k] = H(0)$ nên biểu thức trên được viết là:

$$m_y = H(0).m_x \qquad (7.108)$$

Trong đó H(0) là độ lợi dc của bộ lọc.

7.9.2. Tương quan chéo vào/ra

Tương quan chéo vào/ra được định nghĩa bởi:

$$R_{xy}[m] = E\{x[n].y[n+m]\} = E\left\{x[n].\sum_{j=-\infty}^{\infty} h[j].x[n+m-j]\right\} \qquad (7.109)$$

$$\Leftrightarrow R_{xy}[m] = \sum_{j=-\infty}^{\infty} h[j]E\{x[n]x[n+m-j]\} \qquad (7.110)$$

$$\Leftrightarrow R_{xy}[m] = \sum_{j=-\infty}^{\infty} h[j].R_{xy}[m-j] \qquad (7.111)$$

Tương quan chéo vào/ra được dùng để triển khai nhiều bộ ước tính hiệu năng. Một trong số chúng là bộ ước tính tỉ số tín hiệu trên tạp âm SNR tại một điểm trong hệ thống sẽ được triển khai ở chương 8.

7.9.3. Hàm tự tương quan đầu ra

Theo định nghĩa, hàm tự tương quan đầu ra hệ thống tại trễ *m* là $E\{y[n].y[n+m]\}$. Dẫn đến:

$$R_{yy}[m] = E\{y[n].y[n+m]\} = E\left\{\sum_{j=-\infty}^{\infty} h[j].x[n-j].\sum_{k=-\infty}^{\infty} h[k].x[n+m-k]\right\} \qquad (7.112)$$

$$\Leftrightarrow R_{yy}[m] = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} h[j].h[k].E\{x[n-j].x[n+m-k]\} \qquad (7.113)$$

Ta lại giả định *x[n]* là *dừng* để tiếp tục. Nếu *x[n]* là *dừng*, thì hàm tự tương quan $E\{x[n-j].x[n-k+m]\}$ chỉ phụ thuộc vào trễ *m-k+j* và:

$$R_{yy}[m] = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} h[j].h[k].R_{xx}(m-k+j) \tag{7.114}$$

Trong đó R_{xx} *[m]* là hàm tự tương quan của *x[n]* tại trễ *m*. Đáng tiếc, ở dạng tổng quát biểu thức R_{yy} *[m]* không thể đơn giản hơn nữa nếu không biết rõ về đặc tính thống kê của x[n]. Trừ khi x[n] là chuỗi tương quan delta (trắng). Lấy biến đổi Fourier rời rạc cả 2 vế của (7.114) nhận được quan hệ PSD vào/ra là:

$$S_y(f) = S_x(f).|H(f)|^2 \tag{7.115}$$

Đã được dùng để tạo các chuỗi tương quan.

Nếu chuỗi đầu vào là tương quan delta (nghĩa là tạp âm trắng), thì theo định nghĩa:

$$R_{xx}[m] = E\{x[n].x[n+m]\} = \begin{Bmatrix} \sigma_x^2 \text{ khi } m=0 \\ 0 \text{ khi } m \neq 0 \end{Bmatrix} = \sigma_x^2.\delta[m] \tag{7.116}$$

Thay vào (7.114) ta được:

$$R_{yy}[m] = \sigma_x^2 \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} h[j].h[k].\delta[m-k+j] = \sigma_x^2 \sum_{j=-\infty}^{\infty} h[j]h[m+j] \tag{7.117}$$

Trong đó, ta đã dùng tính chất dịch của hàm deta

7.9.4. Phương sai vào/ra

Theo định nghĩa, phương sai đầu ra hệ thống là $R_{yy}[0] = E\{y^2[n]\}$. Trường hợp tổng quát thì phương sai đầu ra tuân theo (7.114) với $m = 0$. Kết quả là:

$$\sigma_y^2 = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} h[j].h[k].R_{xx}[j-k] \tag{7.118}$$

Nếu *x[n]* là chuỗi tương quan delta (tạp âm trắng), thì tìm được σ_y^2 bằng cách đặt $m = 0$ trong (7.117). Ta có:

$$\sigma_y^2 = \sigma_x^2 \sum_{j=-\infty}^{\infty} h^2[j] \tag{7.119}$$

Kết quả này sẽ được dùng khi nghiên cứu băng thông tạp âm tương đương trong chương 10.

Chương 8
HẬU XỬ LÝ

8.1. Mở đầu

Chương 8 trình bày ngắn ngọn chủ đề quan trọng là hậu xử lý. Vai trò của bộ hậu xử lý là thao tác dữ liệu được tạo ra bởi mô phỏng thành dạng hữu hiệu. Các bộ hậu xử lý thường có tính đồ họa mạnh vì hiển thị trực quan dễ hiểu hơn nhiều so với liệt kê dạng số, nên nó là đầu ra dữ liệu phổ biến nhất từ chương trình mô phỏng. Chẳng hạn để thể hiện kết quả mô phỏng BER với cùng lượng tin, sử dụng đồ thị BER truyền đạt thông tin nhanh hơn và dễ hiểu hơn nhiều so với sử dụng bảng số. Nói cách khác, hậu xử lý cho phép ta trực quan hóa tín hiệu và hệ thống trong các miền xử lý như: miền thời gian, miền tần số, miền z,... cho phép ta quản lý và kiểm soát tiến trình hoạt động của hệ thống một cách tĩnh và động.

Các thường trình hậu xử lý có mức độ phức tạp tính toán đáng kể. Trong khi một số bộ hậu xử lý chỉ lấy dữ liệu được tạo ra bởi mô phỏng, sau đó định dạng dữ liệu để tạo đầu ra đồ hoạ phù hợp. Ví dụ, thường trình tạo đồ thị xác suất lỗi bit (hoặc xác suất lỗi ký hiệu), P_E là hàm của E_b/N_0. Các giá trị của P_E cùng với các giá trị của E_b/N_0 được tạo ra bởi mô phỏng, được qua bộ hậu xử lý ở dạng file dữ liệu. Các bộ hậu xử lý chỉ định dạng dữ liệu và tạo đồ thị cần thiết. Các ví dụ khác về các bộ hậu xử lý là tạo đầu ra đồ hoạ với mức độ xử lý tối thiểu để hiển thị các dạng sóng tín hiệu, các biểu đồ mắt và các biểu đồ tán xạ.

Mặt khác, một số thường trình hậu xử lý bao gồm cả việc xử lý tín hiệu. Đa số chúng đều liên quan đến một số loại ước tính: từ mức đơn giản như tạo ra hoành đồ, là một bộ ước tính hàm mật độ xác suất, cho đến mức phức tạp hơn là các bộ ước tính trễ thời gian, tỷ số tín hiệu trên tạp âm và mật độ phổ công suất. Các ví dụ khác được xét trong chương này gồm ánh xạ tỉ số lỗi ký hiệu kênh thành tỉ số lỗi bit được giải mã đối với hệ thống sử dụng mã hóa kiểm soát lỗi. Liệt kê hết các hoạt động hậu xử lý là không thể, và chương này chỉ đề cập một vài ví dụ điển hình. Tất cả những hoạt động này đều được xem là hậu xử lý vì chúng sử dụng dữ liệu được tạo ra bởi mô phỏng, và chỉ được thực hiện sau khi việc mô phỏng đã hoàn tất.

Muốn vậy chương được tổ chức và thực hiện như sau:

Xét chủ đề hậu xử lý trong mô phỏng bộ điều chế $\pi/4$ DQPSK nhằm trình bày quá trình tạo: đồ thị dạng sóng, chòm sao tín hiệu, biểu đồ mắt, cũng như việc triển khai các bộ ước tính dựa vào dữ liệu mô phỏng. Theo đó:

Trước hết là, xét hoành đồ là bộ ước tính hàm mật độ xác suất. Ở dạng tổng quát, trong khi hoành đồ là bộ ước tính pdf chệch và không kiên định, thì ta sẽ thấy rằng nếu có đủ dữ liệu khả dụng thì cả độ chệch và phương sai đều có thể được bỏ qua.

Tiếp theo là, xét các phiên bản của periodogram như một bộ ước tính PSD của tín hiệu. Ta sẽ thấy rằng, trong khi phương sai của periodogram cơ bản thường lớn ở mức không thể chấp nhận được, thì phương sai đó có thể được giảm bằng cách lấy trung bình các periodogram của các đoạn dữ liệu được cửa sổ hóa. Quá trình phân đoạn hóa và trung bình hóa các periodogram dẫn đến phải dung hoà giữa phương sai và độ phân giải của bộ ước tính.

Tiếp nữa là, xét bộ ước tính để cho phép ước tính: độ lợi của hệ thống, thời gian trễ, tỷ số tín hiệu trên tạp âm SNR. Các kỹ thuật được triển khai ở đây sẽ được sử dụng trong các chương sau khi xét các kỹ thuật bán giải tích.

Chủ đề cuối cùng là, ước tính xác suất lỗi giải mã trong các hệ thống có sử dụng mã hóa kiểm soát lỗi. Kỹ thuật ở đây là dùng mô phỏng để xác định tỉ lệ ký hiệu kênh và sau đó dùng các kỹ thuật giới hạn để ánh xạ các tỉ số lỗi ký hiệu SER kênh thành tỉ số lỗi bit BER được giải mã.

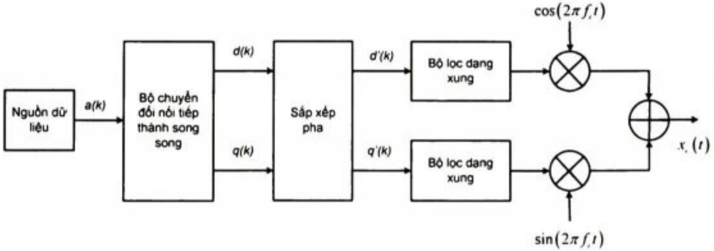
8.2. Kỹ thuật đồ hoạ cơ bản

Để minh hoạ các kỹ thuật đồ họa trong bộ hậu xử lý, các khái niệm được xét bên trong ngữ cảnh của một hệ thống minh họa. Tất nhiên, việc lựa chọn hệ thống minh họa là bất kỳ. Ở đây ta xét cho trường hợp điều chế QPSK vi sai π/4 (ký hiệu π/4 DQPSK) bởi lẽ nó có một số đặc tính được sử dụng trong nhiều hệ thống vô tuyến.

8.2.1. Hệ thống truyền dẫn π/4 DQPSK

Sơ đồ khối máy phát π/4 DQPSK được minh hoạ ở hình 8.1. Đầu ra của nguồn dữ liệu được giả định là một chuỗi ***a*** có dạng:

$$a(1)a(2)a(3)a(4)\ldots a(k)\ldots$$



Hình 8.1: Máy phát $\pi/4$ DQPSK

Bộ chuyển đổi nối tiếp thành song chuyển các ký hiệu có chỉ số lẻ lên kênh đồng pha (trực tiếp) và các ký hiệu có chỉ số chẵn lên kênh vuông pha. Vì vậy:

$$a(1)a(2)a(3)a(4)\ldots a(k)\ldots = d(1)q(1)d(2)q(2)\ldots d\left(\frac{k+1}{2}\right).q\left(\frac{k}{2}+1\right)\ldots$$

Tín hiệu phát được cho bởi:

$$x_c(t) = A\cos\left[2\pi f_c t + \underbrace{\theta(k)}\right],\ (k-1)T_s < t < kT_s \tag{8.1}$$

Trong đó T_s là chu kỳ ký hiệu. Độ lệch pha của tín hiệu phát được xác định bởi các giá trị của *d(k)* và *q(k)* cũng như độ lệch pha $\theta(k-1)$, là độ lệch pha trong chu kỳ ký hiệu trước. Tất

nhiên, sự phụ thuộc vào chu kỳ ký hiệu trước, làm cho $\pi / 4$ **DQPSK** là một kỹ thuật điều chế vi sai. Quan hệ giữa $\theta(k)$ và $\theta(k-1)$ là:

$$\theta(k) = \theta(k-1) + \phi(k) \tag{8.2}$$

Trong đó $\phi(k)$ là hàm của *d(k)* và *q(k)*, và được định rõ trong bảng 8.1. Các pha phát được tạo ra trong bộ sắp xếp pha được cho ở hình 8.1. Bộ sắp xếp pha sử dụng *d(k)*, *q(k)*, và $\theta(k-1)$ để tạo các giá trị mới *d'(k)* và *q'(k)*, sao cho tín hiệu phát có pha phù hợp. Sau khi định dạng xung, các tín hiệu kênh đồng pha và vuông pha được dịch lên tần số truyền dẫn f_c.

Bảng 8.1: Các dịch pha vi sai cho π/4 DQPSK

Các ký hiệu thông tin *d(k)* và *q(k)*	Dịch pha vi sai *ϕ(k)*
1 1	π/4
0 1	3π/4
0 0	-3π/4
1 0	-π/4

Như một ví dụ, giả sử đầu ra của nguồn dữ liệu là chuỗi nhị phân:

$$\underbrace{00}\ \underbrace{10}\ \underbrace{11}\ \underbrace{01}\ \underbrace{11}\ldots \tag{8.3}$$

Và pha ban đầu được định rõ là $\theta(0) = 0$. Vì 2 ký hiệu dữ liệu đầu tiên là 00, nên theo bảng 8.1 ta có $\phi(1) = -3\pi/4$. Từ (8.2) ta có:

$$\theta(1) = 0 - \frac{3\pi}{4} = -\frac{3\pi}{4}$$

2 ký hiệu dữ liệu tiếp theo là 10, do đó $\phi(2) = -\pi/4$. Từ (8.2) ta được:

$$\theta(2) = -\frac{3\pi}{4} - \frac{\pi}{4} = -\pi$$

2 ký hiệu dữ liệu tiếp theo là 11. Vì vậy $\phi(3) = \pi/4$ Từ (8.2) cho ta:

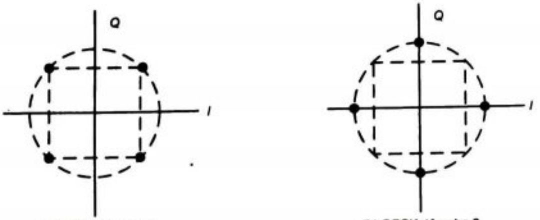
$$\theta(3) = -\pi + \frac{\pi}{4} = -\frac{3\pi}{4}$$

Tương tự $\phi(4) = 3\pi/4$. Từ (8.2) ta được:

$$\theta(4) = -\frac{3\pi}{4} + \frac{3\pi}{4} = 0$$

Tương tự $\phi(5) = \pi/4$ cho ta:

$$\theta(5) = 0 + \frac{\pi}{4} = \frac{\pi}{4}$$

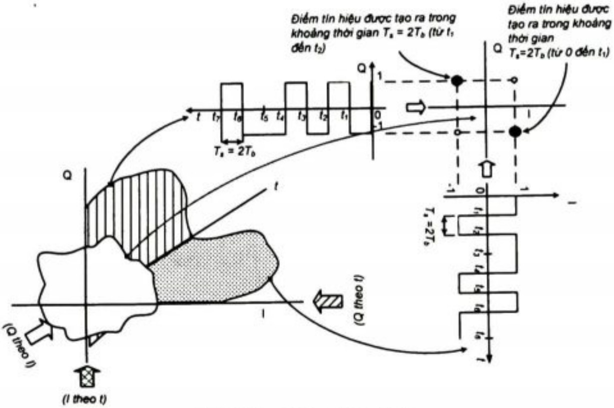


(a) QPSK: tập pha 1 (b) QPSK: tập pha 2

Hình 8.2: Các chòm sao tín hiệu đối với π/4 DQPSK

Quan sát nhanh thấy rõ, $\theta(1)$, $\theta(3)$, $\theta(5)$ là các pha từ chòm sao tín hiệu QPSK thứ nhất được minh hoạ trong hình 8.2 (a), và $\theta(2)$, $\theta(4)$ là các pha từ chòm sao tín hiệu QPSK thứ hai được minh hoạ trong hình 8.2 (b). Vì vậy, $\pi/4$ DQPSK hoạt động bằng cách phát các điểm tín hiệu phát từ các chòm sao tín hiệu QPSK một cách luân phiên nhau trong đó cứ 2 chòm sao tín hiệu QPSK được chuyển chỗ bằng cách quay pha $\pi/4$. Mặc dù điều này được minh họa bằng cách sử dụng một chuỗi dữ liệu cụ thể, nhưng ta thấy rằng kết quả là tổng quát, vì các pha vi sai chỉ lấy các giá trị $\pm\pi/4$ hoặc $\pm 3\pi/4$.

8.2.2. Dạng sóng, biểu đồ mắt và biểu đồ tán xạ



Hình 8.3: Hệ toạ độ 3 chiều

Trước khi giải thích các chương trình cơ bản để vẽ các dạng sóng, các biểu đồ mắt, và các đồ thị tán xạ, ta minh hoạ quan hệ giữa những biểu đồ này. Giả sử ta triển khai một hệ toạ độ 3 chiều được cho ở hình 8.3 có các trục được đánh nhãn tương ứng. Lưu ý rằng, hình thành 3 mặt phẳng, mỗi mặt phẳng chứa 2 trục được tạo ra bởi các trục: (I và t), (Q và t), (Q và I). *Nếu*: *(i)* tín hiệu kênh đồng pha $x_d(t)$ được vẽ trên mặt phẳng (I, t); *(ii)* tín hiệu kênh vuông pha $x_q(t)$

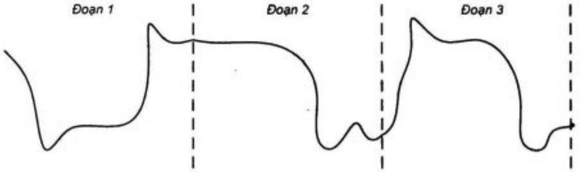
được vẽ trên mặt phẳng (Q,*t*), *thì* tín hiệu 3 chiều **được thông số hóa theo** *t* (hàm của *t*) **được tạo** ra. Chiếu tín hiệu này lên một không gian con cho trước (I,*t*), (Q,*t*), hoặc (I,Q) tạo ra $x_d(t)$, $x_q(t)$, hay biểu đồ tán xạ (*vẽ* $x_q(t)$ *là một hàm của* $x_d(t)$). Điều này được minh hoạ trong hình 8.3. Quan sát từ phía phải, cho thấy tín hiệu vuông pha $x_q(t)$ là hàm của t, được vẽ trên mặt phẳng (Q,*t*). Tương tự, quan sát từ dưới, cho thấy tín hiệu kênh đồng pha $x_d(t)$ là hàm của *t* (được vẽ trên mặt phẳng (I,*t*)). Nhìn xuống trục thời gian sao cho trục thời gian trở thành một điểm lộ ra đồ thị tán xạ, $x_q(t)$ là hàm của $x_d(t)$, được vẽ trên mặt phẳng (Q,I) .

Trong khi ảnh 3 chiều (Q, I, *t*) là hiếm khi được tạo ra trong thực tế, nhưng việc trực quan hóa (Q, I, *t*) là một công cụ học tập tốt và cho thấy rõ mối quan hệ giữa $x_d(t)$, $x_q(t)$ và biểu đồ tán xạ.

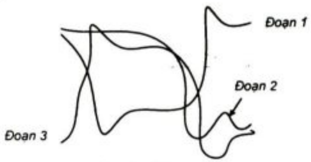
Biểu đồ mắt

Biểu đồ mắt cho phép đánh giá *định tính* hiệu năng hệ thống. Một mắt mở và sáng thể hiện hiệu năng tốt, ngược lại hiệu năng kém. Ngoài ra, kích thước của mắt liên quan đến yêu cầu độ chính xác của bộ đồng bộ ký hiệu. Trong khi biểu đồ mắt không cho đánh giá *định lượng* hiệu năng hệ thống, thì khó để quan niệm một hệ thống hiệu năng tốt có biểu đồ mắt xấu.

Quá trình hình thành biểu đồ mắt được minh hoạ ở hình 8.4. Ba đoạn dạng sóng trong đó mỗi đoạn tương ứng với mỗi chu kỳ ký hiệu. Dạng sóng tương ứng với 3 ký hiệu dữ liệu được minh hoạ trong hình 8.4(a). Giả sử rằng dạng sóng này được hiển thị trên máy hiện sóng và máy hiện sóng được đồng bộ với dạng sóng tại các điểm đường sọc thẳng. Kết quả sẽ là biểu đồ mắt 3 đoạn được minh hoạ trong hình 8.4 (b).



(a) 3 đoạn (60 mẫu) của dạng sóng

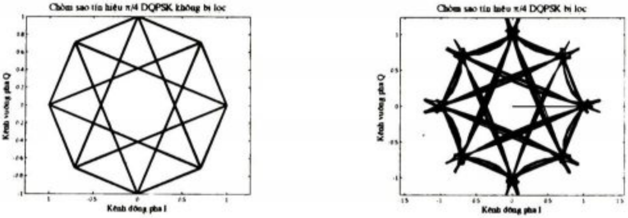


(b) Biểu đồ mắt 3 đoạn

Hình 8.4: Tạo biểu đồ mắt

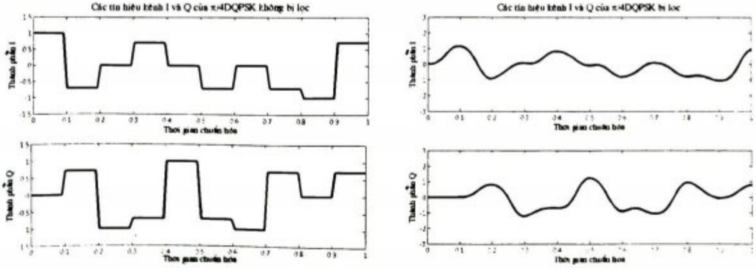
Ví dụ 8.1: Ví dụ này, sẽ tạo và hiển thị một vài tín hiệu quan trọng trong hệ thống $\pi/2$ DPSK. Chương trình Matlab mô phỏng hệ thống và tạo đồ hoạ được cho ở file **NVD8_pi4sim** ở Phụ lục 8A. Chạy chương trình **NVD_pi4sim** tại dấu nhắc cửa sổ lệnh Matlab. Từ menu này chọn một trong 7 lựa chọn sau:

1. Chòm sao tín hiệu π/4 DQPSK không bị lọc
2. Biểu đồ mắt tín hiệu π/4 DQPSK không bị lọc
3. Chòm sao tín hiệu π/4 DQPSK bị lọc
4. Biểu đồ mắt tín hiệu π/4 DQPSK bị lọc
5. Các tín hiệu vuông pha và đồng pha không bị lọc
6. Các tín hiệu vuông pha và đồng bị lọc
7. Thoát chương trình mô phỏng

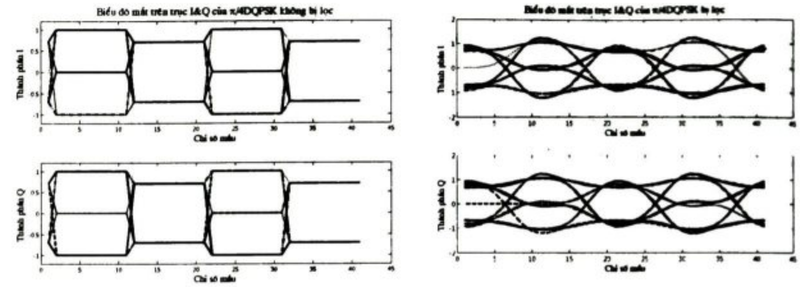


Hình 8.5: Chòm sao tín hiệu π/4 DQPSK không bị lọc và bị lọc

Để được rõ, cần phải nghiên cứu mã chương trình mô phỏng trong Phụ lục 8A, ở đó minh hoạ nhiều thủ tục hậu xử lý. Ngoài ra mã chương trình được sử dụng để tạo các đồ thị khác nhau trong bộ hậu xử lý của các chương trình mô phỏng khác. Ở đây minh hoạ 3 kết quả đáng quan tâm. Hình 8.5, 8.6 và 8.7 minh hoạ biểu đồ tán xạ (chòm sao tín hiệu), các tín hiệu kênh vuông pha và kênh đồng pha, và các biểu đồ mắt trên kênh vuông pha và đồng pha. Lưu ý rằng, bằng cách mường tượng tín hiệu ba chiều trong không gian (I,Q,t) thấy rõ mối quan hệ giữa các hình 8.5 và 8.7. Để được rõ hơn về mức độ ảnh hưởng của tham số hệ thống lên các kết quả quan sát, ta nên thay đổi các tham số bộ lọc và các tham số khác như số ký hiệu mô phỏng, lấy mẫu trên ký hiệu.



Hình 8.6: Các thành phần I & Q của tín hiệu π/4 DQPSK bị lọc và bị lọc



*Hình 8.7: **Biểu đồ mắt của tín hiệu π/4 DQPSK không bị lọc và bị lọc***

8.3. Ước tính

Nhiều thường trình ước tính hiệu năng dựa trên dữ liệu được tạo ra bởi chương trình mô phỏng. Ở đây ta xét một số thường trình ước tính căn bản nhất. Chi tiết hóa cho vấn đề này được cho ở phần Phụ lục 7B2, ở đó được trình bày súc tích và rất trực quan dễ hiểu ở dạng các định nghĩa và các biểu thức kết quả.

8.3.1. Hoành đồ

Nếu có sẵn một tập các mẫu của một quá trình ngẫu nhiên (từ mô phỏng hoặc đo kiểm), thì ta có thể tạo ra hoành đồ từ tập các mẫu, là một bộ ước tính hàm mật độ xác suất pdf cơ bản. Hoành đồ được tạo ra bằng cách nhóm dữ liệu, toàn bộ N mẫu dữ liệu vào B **bin** (B ngăn). Giả sử mỗi bin đều có cùng độ rộng W, và tâm của mỗi bin là b_i. Vì vậy, một mẫu *x[n]* cho trước rơi vào bin thứ *i* nếu:

$$b_i - \frac{W}{2} < x[n] \le b_i + \frac{W}{2} \tag{8.4}$$

Đại lượng quan tâm là N_i, thể hiện số mẫu rơi vào bin thứ *i*. Rõ ràng:

$$N = \sum_{i=1}^{B} N_i \tag{8.5}$$

Ta ký hiệu *Count{N:R}* để thể hiện số lượng mẫu trong tập *N* mẫu tổng rơi vào bin hoành đồ được xác định bởi *R*. Vì vậy:

$$N_i = Count\left\{N : \underbrace{b_i - \frac{W}{2} < x[n] \le b_i + \frac{W}{2}}_{R}\right\} \tag{8.6}$$

Đồ thị dạng thanh, trong đó chiều cao của mỗi thanh tỷ lệ với N_i, và mỗi thanh được trung tâm tại b_i. Để là một bộ ước tính pdf, thì hoành đồ được tỷ lệ hóa sao cho toàn bộ diện tích đó bằng 1. Muốn vậy, ta chia N_i cho *NW*, khi này chiều cao của mỗi thanh là *N/NW*. Diện tích của thanh, A_i, thể hiện bin hoành đồ thứ *i*, được tìm thấy bằng cách nhân chiều cao với độ rộng *W*. Vì vậy:

$$A_i = \left(\frac{N_i}{N.W}\right).W = \frac{N_i}{N} \tag{8.7}$$

Lưu ý rằng, A_i thể hiện tần suất tương đối của bin thứ *i*. Diện tích tổng là:

$$\sum_{i=1}^{B} \frac{N_i}{N} = 1 \tag{8.8}$$

Như yêu cầu đề ra nếu muốn sử dụng hoành đồ để biểu diễn hàm mật độ xác suất pdf

Lưu ý rằng, mỗi bin của hoành đồ biểu diễn pdf trên độ rộng *W*. Với một số điểm bên trong một bin cho trước, thì bộ ước tính pdf sẽ *không bị chệch*. Tuy nhiên, trên hầu hết khoảng được định nghĩa bởi một bin hoành đồ cho trước, thì bộ ước tính sẽ *bị chệch*. Dễ dàng thấy rằng, bằng cách khai triển hàm pdf $f_X(x)$ theo chuỗi Taylor xung quanh tâm của bin thứ *i*, thì độ chệch là:

$$E\left\{f_X(b_i) - \hat{f}_X(b_i)\right\} \approx \left.\frac{d^2 f_X(x)}{dx^2}\right|_{x \to b_i} \frac{W^2}{24} \tag{8.9}$$

Vì vậy, độ chệnh chỉ có thể được giảm bằng cách giảm độ rộng bin *W*.

Cũng có thể thấy rằng phương sai của bộ ước tính là:

$$\operatorname{var}\left\{\hat{f}_X(b_i)\right\} \approx \frac{1}{N.W} f_X(b_i)\left[1 - \underbrace{W.f_X(b_i)}_{\substack{\text{xác suất của sự kiện} \\ \text{một mẫu cho trước rơi vào bin } b_i}}\right] \tag{8.10}$$

Lưu ý rằng, $W.f_X(b_i)$ là xác suất của sự kiện mà một mẫu cho trước rơi vào bin b_i. Vì nó là xác suất, nên $W.f_X(b_i) \le 1$ là nhỏ hơn 1 và thường nhỏ hơn 1 rất nhiều. Vì vậy:

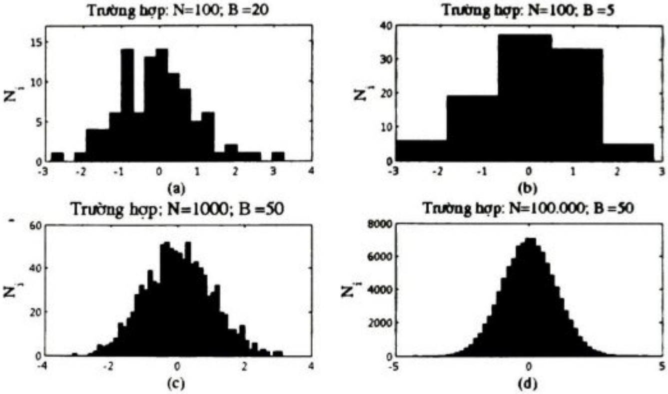
$$\operatorname{var}\left\{\hat{f}_X(b_i)\right\} \approx \frac{1}{N.W} f_X(b_i) \tag{8.11}$$

Ta thấy rằng, với *N* cố định, việc tăng *W* sẽ làm giảm phương sai của bộ ước tính, nhưng lại làm tăng độ chệch, được thấy từ (8.9). Vì vậy, ta mong muốn *W* nhỏ và *NW* lớn. Vì vậy, phải có quan hệ giữa *N* và *W*. Chẳng hạn, nếu $W = 1/\sqrt{N}$ và $NW = \sqrt{N}$, thì $W \to 0$ khi $N \to \infty$ *và* $NW \to \infty$ khi $N \to \infty$. Trong trường hợp này, với *N* đủ lớn thì pdf có độ chệch và phương sai là không đáng kể. Ví dụ 8.2 minh hoạ hoành đồ đối với những lựa chọn *N* và *W* khác nhau.

Ví dụ 8.2: Giả sử, ta tạo *N* mẫu của một biến ngẫu nhiên Gausơ phương sai đơn vị trung bình 0. Hàm pdf được ước tính bằng cách xây dựng một hoành đồ có *B* bin. Trong ví dụ này, ta sẽ khảo sát ảnh hưởng của việc thay đổi giá trị của *N* và *B*. Chương trình Matlab thực hiện bài toán này được cho ở file **NVD8_hist.m** (có trong Phụ lục 8A).

Kết quả chạy chương trình này được minh hoạ ở hình 8.8. Hình 8.8(a) minh hoạ các kết quả khi $N = 100$ và $B = 20$, cho thấy hoành đồ không được định rõ lắm vì $N/B = 5$ là quá nhỏ để tạo ra bộ ước tính tin cậy cho một số mẫu rơi vào bin cho trước. Hình 8.8 (b) minh hoạ kết quả khi $N = 100$ và $B = 5$. Trong khi tỷ số N/B là quá lớn thì số bin là quá nhỏ. Lưu ý rằng khi có định *N* thì biểu đồ thống kê có $B = 5$ sẽ biểu lộ độ chệnh lớn hơn so với trường hợp $B = 20$.

Hình 8.8(c) minh hoạ kết quả khi $N = 1000$ và $B = 50$. Trong khi cho ta bộ ước tính tốt hơn thì tỷ số N/B lại quá nhỏ và ta tăng N. Hình 8.8 (d) minh hoạ kết quả cho $N = 100000$ và $B = 50$, tạo ra $N/B = 2000$ và biểu đồ thống kê có hình dạng Gausơ được dự đoán trước. Nếu pdf cơ bản có hình dạng phức tạp, thì phải cần đến số lượng mẫu rất lớn nếu muốn có được hoành đồ là bộ ước tính chính xác của pdf.



Hình 8.8: Khảo sát hoành đồ của với biến ngẫu nhiên Gausơ với các giá trị của N và B

8.3.2. Ước tính mật độ phổ công suất

Một hoạt động hậu xử lý khác thường được dùng trong nghiên cứu mô phỏng là ước tính mật độ phổ công suất PSD của một tín hiệu tại một điểm trong hệ thống. Đây là một nhiệm vụ khó hơn. Thường thì, dạng sóng quan tâm là hàm mẫu của quá trình ngẫu nhiên. Dẫn đến trường hợp PSD tại một giá trị tần số cho trước f_l là một biến ngẫu nhiên. Vì vậy, nảy sinh bài toán giảm thiểu phương sai của ước tính phổ $\ddot{S}(f_1)$. Tồn tại nhiều tài liệu và bài báo được viết cho chủ đề này, và nhiều kỹ thuật đã được triển khai để ước tính PSD. Trong phần này, ta chỉ xét các kỹ thuật cơ bản nhất. Các kỹ thuật được xét ở đây dựa vào thuật toán FFT, và được dùng trong mô phỏng.

Periodogram

Thuật toán ước tính PSD đơn giản nhất, nhanh nhất, và thường được sử dụng là periodogram, được định nghĩa bởi:

$$\ddot{S}(kf_\Delta) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x[n].e^{-j2\pi kf_\Delta n}\right|^2 = \frac{1}{N}\left|I_N(kf_\Delta)\right|^2, \quad k = 0, 1, ..., N-1 \tag{8.12}$$

Trong đó: N là tổng số mẫu có trong bản ghi dữ liệu, và $I_N(kf_\Delta)$ là FFT N điểm của dữ liệu trong đó ước tính PSD tại tần số $f = kf_\Delta$ sẽ được tính toán. Hiệu quả tính toán của periodogram từ việc sử dụng FFT để tạo ước tính PSD. Kết quả ta được N ước tính trong miền

tần số với độ phân giải là $f_\Delta = f_s / N$, trong đó f_s là tần số lấy mẫu liên quan đến các điểm trong miền thời gian để thực hiện ước tính phổ. Lưu ý rằng, f_s trong trường hợp này không phải lúc nào cũng là tần số lấy mẫu liên quan đến mô phỏng. Chẳng hạn, nếu trong một mô phỏng cho trước một tín hiệu được lấy mẫu quá mức, thì các mẫu đó có thể được giảm bớt trước khi thực hiện ước tính phổ (xem kỹ thuật tăng và giảm mẫu ở chương 3).

Khó khăn đối với periodogram, nó là bộ ước tính PSD *chệch* và *không kiên định* tại tần số *f*. Trong nhiều ứng dụng, phương sai của periodogram là không thể chấp nhận được. Thực tế, do bản ghi số liệu là hữu hạn, nên không thể tránh khỏi kết quả bị chệch. Tuy nhiên, khi *N* đủ lớn thì độ chệch có thể bỏ qua. Vì vậy, khó khăn chủ yếu là do phương sai lớn. Giả sử, các mẫu dữ liệu *x[n]* là độc lập, thì phương sai của ước tính phổ tại tần số *f* là:

$$\operatorname{var}\left(\hat{S}(f)\right) = \sigma_x^4 \left\{ 1 + \left(\frac{\sin[2\pi fN]}{N.\sin[2\pi f]} \right)^2 \right\} \tag{8.13}$$

Trong đó σ_x^2 là phương sai của các mẫu dữ liệu *x[n]*. Ta thấy rằng, phương sai của $\hat{S}(f)$ không tiến tới 0 khi $N \to \infty$, và khi *N* đủ lớn thì phương sai của ước tính phổ trở nên độc lập với tần số. Tuy nhiên periodogram, nó vẫn hữu hiệu để *"tìm kiếm nhanh"* PSD.

Periodogram với cửa sổ dữ liệu

Nếu cửa sổ dữ liệu chưa được định rõ, thì cửa sổ chữ nhật mặc định được sử dụng. Với cửa sổ chữ nhật, mỗi giá trị mẫu *x[n]* được nhân với *w[n]* = 1 khi $0 \le n \le N-1$. Ảnh hưởng của cửa sổ chữ nhật là lấy tổng chập các mẫu dữ liệu *x[n]*, với biến đổi Fourier của *w[n]*, nó có phổ biên độ là $\sin(\pi Nf)/N\sin(\pi f)$. Khi xét trong miền tần số, thì cấu trúc búp phụ của cửa sổ chữ nhật làm mất đáng kể phổ, dẫn đến làm méo và làm giảm dải động của phổ được ước tính.

Khi sử dụng của sổ dữ liệu bất kỳ, thì ước tính phổ $\hat{S}(kf_\Delta)$ có dạng:

$$\tilde{S}(kf_\Delta) = \frac{1}{U} \left| \sum_{n=0}^{N-1} x[n].w[n].e^{-j2\pi kf_\Delta n} \right|^2 \tag{8.14}$$

Trong đó *U* là năng lượng trong cửa sổ dữ liệu, được cho bởi:

$$U = \sum_{n=0}^{N-1} w^2[n] \tag{8.15}$$

Lưu ý rằng, với cửa sổ chữ nhật trong đó $w[n] = 1,\ \forall n$, thì *U* = *N* và nhận được các kết quả (8.12). Việc lựa chọn cửa sổ dữ liệu thể hiện một số dung hoà. Cửa sổ chữ nhật lý tưởng phải có khoảng thời gian hữu hạn sao cho biến đổi Fourier của dữ liệu đó $I_N(kf_\Delta)$ được ước tính một cách chính xác. Ngoài ra, biến đổi Fourier của bản ghi dữ liệu được ước tính đó phải không bị ảnh hưởng xấu bởi hàm của sổ. Do tính đối ngẫu của biến đổi Fourier là, phép nhân trong miền thời gian tương ứng phép tích chập trong miền tần số, và vì vậy chỉ khi lấy tích chập với hàm xung kim thì mới không làm thay đổi phép biến đổi đó, nên hàm cửa sổ lý tưởng là một ***xung kim*** trong miền tần số (lưu ý rằng, phổ của một xung kim là vô hạn, và ngược lại

biến đổi Fourier ngược của một xung kim tần số là vô hạn). Vì biến đổi Fourier của xung kim là vô hạn nên tồn tại các yêu cầu đối nghịch. Vì vậy, cần phải tìm của sổ dữ liệu trong miền tần số để lộ ra búp chính hẹp quanh tần số $f = 0$ và các búp phụ bị suy hao lớn.

Các periodogram phân đoạn

Kỹ thuật phổ dụng để giảm phương sai liên quan với periodogram là chia bản ghi dữ liệu N mẫu thành K đoạn với mỗi đoạn gồm M mẫu. FFT được tính toán cho mỗi đoạn, sau đó lấy trung bình các kết quả. Quá trình lấy trung bình làm giảm phương sai của ước tính phổ. Các đoạn có thể sẽ chồng lấn nhau hoặc không chồng lấn nhau. Nếu các đoạn không chồng lấn nhau thì $K = M/N$, ngược lại thì $K > M/N$.

Periodogram của đoạn dữ liệu thứ i được cho bởi:

$$I_M^{(i)}(kf_\Delta) = \frac{1}{U}\left|\sum_{n=0}^{M-1} x^{(i)}[n].w[n].e^{-j2\pi kf_\Delta n}\right|^2 \quad i = 1, 2, \ldots, K \tag{8.16}$$

Trong đó $x^{(i)}[n]$ là các mẫu trong bản ghi dữ liệu thứ i và $f_\Delta = f_s / M$. Sau đó lấy trung bình K periodograms sẽ tạo bộ ước tính PSD:

$$\hat{S}(kf_\Delta) = \frac{1}{K}\sum_{i=1}^{K} I_M^{(i)}(kf_\Delta),\ \mathrm{k} = 0, 1, \ldots, \mathrm{M} - 1 \tag{8.17}$$

Tất nhiên, bộ ước tính này là chệch bởi vì bản ghi dữ liệu là hữu hạn. Giả sử rằng K periodograms là độc lập:

$$\mathrm{var}\left(\hat{S}(kf_\Delta)\right) = \frac{1}{K}S^2(kf_\Delta) \tag{8.18}$$

dần tới 0 khi $K \to \infty$.

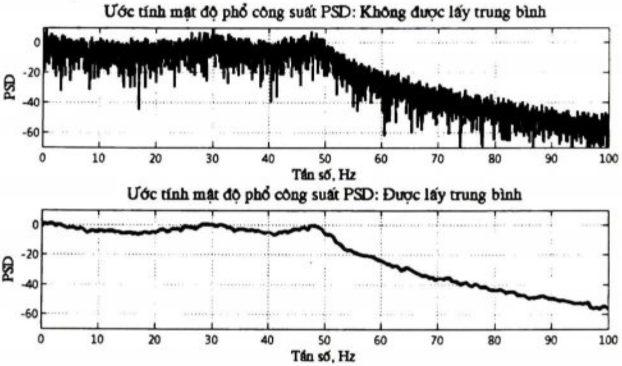
Thấy rõ từ việc so sánh (8.12) và (8.16). Trong khi periodogram được định nghĩa bởi (8.12) có độ phân giải tần số là $f_\Delta = f_s/N$, thì periodogram được định nghĩa bởi (8.16) có độ phân giải tần số là $f_\Delta = f_s/M$. Vì $M<N$ khi $K > 1$, nên khi phân đoạn bản ghi dữ liệu N mẫu ban đầu thì độ phân giải tần số bị suy giảm. Do đó, khi sử dụng kỹ thuật phân đoạn sẽ nảy sinh yêu cầu cần phải dung hoà giữa độ phân giải và phương sai. Cũng vậy, tính hợp lý của (8.18) đòi hỏi K periodogram được sử dụng trong quá trình lấy trung bình là độc lập. Vì mong muốn có giá trị lớn nhất có thể của K với N cố định, nên các đoạn thường bị chồng chập nhau. Thường dùng chồng chập 50%. Tuy nhiên, nếu các đoạn dữ liệu bị chồng chập thì K periodogram không còn độc lập nữa và việc giảm phương sai của bộ ước tính PSD là kém hơn được dự đoán bởi (8.18). Ít nhất từng phần, việc sử dụng cửa sổ dữ liệu giúp cho khôi phục tính độc lập của K đoạn.

Trong khi tồn tại nhiều loại của sổ dữ liệu được dùng trong (8.16), thì cửa sổ Hanning thường được sử dụng để ước tính PSD. Cửa sổ Hanning được định nghĩa bởi

$$w[n] = \begin{cases} \frac{1}{2}\left[1 - \cos\left(\frac{2\pi n}{M-1}\right)\right], & 0 \le n \le M-1 \\ 0, & \text{nếu khác} \end{cases} \tag{8.19}$$

Ví dụ 8.3: Trong ví dụ này ta cho các mẫu độc lập (tạp âm trắng) qua bộ lọc Chebyshev có độ gợn sóng băng thông 5 dB. Bài toán là ước tính PSD tại đầu ra bộ lọc. Chương trình Matlab được cho bởi file **NVD8_PSDSim.m** (có trong Phụ lục 8A) thực hiện bài toán này.

Kết quả chạy chương trình mô phỏng **NVD8_PSDSim.m** được cho ở hình 8.9. Cần lưu ý: *(i)* trường hợp không chồng chéo (hình 8.9 phía trên) phương sai lớn. *(ii)* Phương sai không phụ thuộc vào tần số. Sử dụng 25 đoạn (hình 8.9 phía dưới) làm cho phương sai nhỏ hơn nhiều với mức chi phí độ phân giải tần số được giảm. *(iii)* Độ gợi sóng thông dải 5 dB của bộ lọc Chebyshe khi $K = 25$ rõ ràng hơn nhiều khi $K = 1$.

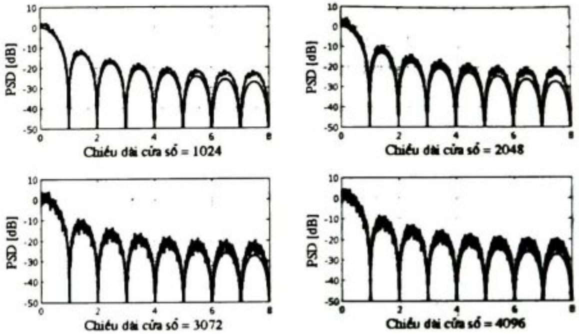


Hình 8.9: Các ước tính PSD: không được lấy trung bình và được lấy trung bình

Ta triển khai bộ ước tính PSD được minh hoạ trong ví dụ 8.3 bằng cách dùng các lệnh Matlab cơ bản. Hộp công cụ xử lý tín hiệu (*Signal Processing Toolbox*) trong Matlab chứa một số thường trình để ước tính PSD. Hai trong số chúng là hàm **psd** và **pwelch**. Để được rõ hơn, ta nên nghiên cứu kỹ cho một số trường hợp cụ thể. Ở đây ta minh hoạ periodogram Welch từ hộp công cụ xử lý tín hiệu.

Ví dụ 8.4: Trong ví dụ này, ta ước tính PSD của tín hiệu QPSK. Với giả thiết là: *(i)* Định dạng xung chữ nhật; *(ii)* lấy mẫu các thành phần đồng pha và vuông pha của tín hiệu QPSK tại tốc độ 16 mẫu/ký hiệu. Mã chương trình Matlab được cho ở file **NVD8_welchp.m** (có trong Phụ lục 8A)

Chạy chương trình mô phỏng **NVD8_welchp.m** ta được các kết quả được minh hoạ ở hình 8.10. Lưu ý rằng, chương trình tạo ra 16×1024 điểm và sử dụng các kích thước cửa sổ (*nwindow*) là 1024, 2048, 3072, 4096. Cũng như ví dụ trước khi *nwindow* lớn tạo ra trung bình nhỏ hơn, dẫn đến PSD được ước tính tỏ ra phương sai lớn hơn. Khi *nwindow* nhỏ sẽ làm giảm phương sai ở mức chi chí độ phân giải giảm. Những xu hướng này có thể được thấy từ hình 8.10.



Hình 8.10: Các ước tính PSD được tạo ra trong ví dụ 8.4

8.3.3. Độ lợi, trễ và tỷ số tín hiệu trên tạp âm

Tỷ số tín hiệu trên tạp âm SNR là hệ số phẩm chất được sử dụng phổ biến để ước lượng hiệu năng hệ thống truyền thông. Kỹ thuật ước tính SNR được trình bày ở đây được xuất phát từ phương pháp đo các lỗi méo kênh trong các hệ thống đo xa băng rộng, tạp âm trong tín hiệu tại một điểm trong hệ thống được xác định là *lỗi trung bình quân phương* MSE giữa tín hiệu thực tế và tín hiệu mong muốn tại điểm đó. SNR được ước tính bằng cách xác định tín hiệu mong muốn (biên độ được tỉ lệ hóa) *và* phiên bản bị trễ của tín hiệu mang thông tin tại đầu vào hệ thống. Trước đây đã tính đến việc ứng dụng kỹ thuật này để giám sát độ tin cậy truyền dẫn dữ liệu điều chế xung mã số và ước tính tỷ số sóng mang trên méo điều chế ký sinh IM trong kênh phi tuyến.

Triển khai lý thuyết cho các tín hiệu thông thấp giá trị thực

Đối với hệ thống tuyến tính bất biến LTIV không méo, thì tín hiệu *y(t)* tại bất kỳ điểm nào trong hệ thống đều là phiên bản biên độ bị tỷ lệ hóa và phiên bản của tín hiệu tham chiếu đầu vào *x(t)* bị trễ. Vì vậy ta có thể viết tín hiệu không méo là:

$$z(t) = A.x(t-\tau) \tag{8.20}$$

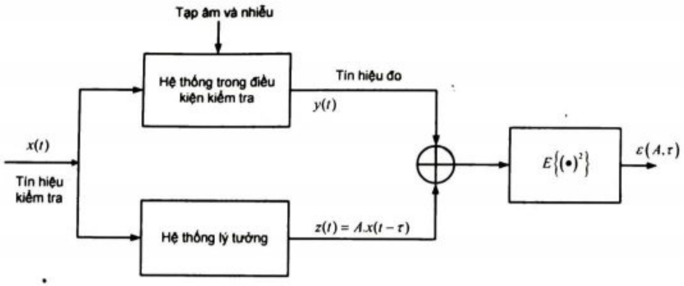
Trong đó A là độ lợi và τ là trễ nhóm đối với điểm trong hệ thống tại đó SNR sẽ được xác định. Lấy *x(t)* là tín hiệu tham chiếu và *y(t)* là tín hiệu đo, thì:

$$y(t) = A.x(t-\tau) + n(t) + d(t) \tag{8.21}$$

Trong đó *n(t)* là tạp âm cộng bên ngoài, *d(t)* là méo bên trong phụ thuộc tín hiệu do hệ thống gây ra (có thể do ISI hoặc tính phi tuyến hệ thống). Sơ đồ khối mô tả quan hệ giữa các tín hiệu trong hệ thống được cho ở hình 8.11.

Công suất tạp âm được định nghĩa là MSE giữa *y(t)* và đầu ra của hệ thống không méo $z(t) = A.x(t-\tau)$. Nghĩa là:

$$\varepsilon(A,\tau) = E\left\{\left[y(t) - A.x(t-\tau)\right]^2\right\} \tag{8.22}$$



Hình 8.11: Các thủ tục kiểm tra để ước tính độ lợi, trễ, SNR

Các ước tính mong muốn đối với A và τ là các giá trị tại đó $\varepsilon(A,\tau)$ được cực tiểu. Biểu thức trên được viết là:

$$\varepsilon(A,\tau)=E\{y^2(t)+A^2x^2(t-\tau)-2Ax(t-\tau)y(t)\} \tag{8.23}$$

Đối với các tín hiệu *dừng*, thì các mô men là độc lập với gốc thời gian. Ngoài ra kỳ vọng toán học của một tổng là tổng các giá trị kỳ vọng. Phương trình (8.23) được viết:

$$\varepsilon(A,\tau)=\underbrace{E\{y^2(t)\}}_{P_y}+A^2\underbrace{E\{x^2(t)\}}_{P_x}-2A\underbrace{E\{x(t)y(t+\tau)\}}_{R_{xy}(\tau)} \tag{8.24}$$

$$\Leftrightarrow \varepsilon(A,\tau)=P_y+A^2P_x-2AR_{xy}(t) \tag{8.25}$$

Trong đó P_x và P_y thể hiện các công suất trung bình của *x(t)* và *y(t)* tương ứng. Thấy rõ, giá trị τ làm cực tiểu $\varepsilon(A,\tau)$ là giá trị của τ, được ký hiệu là τ_m, làm cực đại hàm tương quan chéo giữa *x(t)* và *y(t)*, $R_{xy}(\tau)$. Ta gọi đây là trễ thời gian hệ thống. Độ lợi hệ thống A_m là giá trị A để:

$$\frac{d}{dA}\varepsilon(A,\tau_m)=\frac{d}{dA}\left[P_y+A^2P_x-2AR_{xy}(\tau_m)\right]=0 \tag{8.26}$$

$$\Rightarrow A_m=\frac{R_{xy}(\tau_m)}{P_x} \tag{8.27}$$

Công suất trong tín hiệu lỗi được tìm thấy từ (8.25) với $A = A_m$ và $\tau=\tau_m$. Vì đây là thành phần của *y(t)* trực giao với tín hiệu *x(t)*, nên ta xác định công suất trong tín hiệu lỗi là công suất tạp âm N. Vì vậy:

$$N=P_y-\frac{R_{xy}^2(\tau_m)}{P_x} \tag{8.28}$$

Công suất trong thành phần tín hiệu của *y(t)* là:

$$S = P_z = A_m^2 P_x = \frac{R_{xy}(\tau_m)}{P_x} \tag{8.29}$$

Vì vậy, tỷ số tín hiệu trên tạp âm SNR là:

$$\frac{S}{N} = \frac{R_{xy}^2(\tau_m)}{P_x}\left[\frac{P_x}{P_x P_y - R_{xy}^2(\tau_m)}\right] = \frac{R_{xy}^2(\tau_m)}{P_x P_y - R_{xy}^2(\tau_m)} \tag{8.30}$$

Hệ số tương quan ρ liên quan giữa *x(t)* và *y(t)* được định nghĩa là:

$$\rho = \frac{R_{xy}^2(\tau_m)}{\sqrt{P_x P_y}} \tag{8.31}$$

Với định nghĩa này, SNR tại điểm đo *y(t)* có dạng rất đơn giản là:

$$\frac{S}{N} = \frac{\rho^2}{1-\rho^2} \tag{8.32}$$

Với những hệ thống có các tín hiệu thực, bài toán này được nghiên cứu bởi Turner, Tranter, Eggleston, Jeruchim và Wolfe. Việc làm cực tiểu $\varepsilon(A,\tau)$ hay đồng nghĩa với việc cực đại hóa hàm tương quan chéo $R_{xy}(\tau)$ giữa tín hiệu tham chiếu *x(t)* và tín hiệu đo *y(t)*.

Chương trình Matlab thực hiện bộ hậu xử lý để ước tính độ lợi hệ thống, trễ và SNR được cho bởi file **NVD8_ snrmse.m** (có trong Phụ lục 8A).

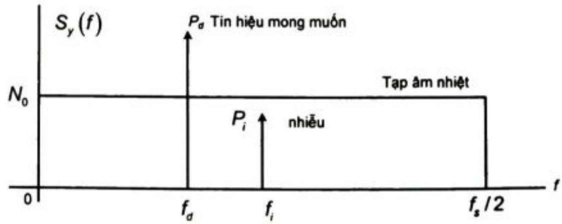
Tại đây, ta xét một ví dụ đơn giản (lưu ý: kỹ thuật được dùng ở đây để ước tính trễ sẽ được dùng trong chương 10 ở đó xét bài toán mô phỏng bán giải tích).

Ví dụ 8.5: Để minh hoạ các kỹ thuật trên, ta giả sử *x(t)* là tín hiệu hình sin:

$$x(t) = A.\sin(2\pi f_d t) \tag{8.33}$$

Và tín hiệu đo là:

$$y(t) = G.A.\sin(2\pi f_d t + \phi) + B.\sin(2\pi f_i t) + \sigma_n n(t) \tag{8.34}$$



Hình 8.12: PSD một phía của tín hiệu đo y(t)

Trong đó *G* là độ lợi hệ thống, *n(t)* là quá trình tạp âm Gausơ trắng phương sai đơn vị trung bình 0 và σ_n là độ lệch chuẩn của quá trình tạp âm cộng. PSD của *y(t)* được minh hoạ ở

hình 8.12, trong đó P_d là công suất tín hiệu (công suất thành phần mong muốn), P_i là công suất nhiễu và N_0 là mật độ phổ công suất một phía của thành phần tạp âm. Từ chương 7 cho thấy, quan hệ giữa N_0 và σ_n là:

$$\sigma_n^2 = N_0 \frac{f_s}{2} \tag{8.35}$$

trong đó f_s là tần số lấy mẫu.

Đối với ví dụ này, tín hiệu tham chiếu được xác định bởi:

$$x(t) = 80.\sin\left[2\pi(2)t\right] \tag{8.36}$$

Tín hiệu tại đầu vào máy thu được giả định là:

$$y(t) = 20.\sin\left[2\pi(2)t - \frac{\pi}{4}\right] + 4.\sin\left[2\pi(8)t\right] + 0{,}8.n(t) \tag{8.37}$$

Trong đó *n(t)* là hàm mẫu của quá trình tạp âm phương sai đơn vị trung bình 0. Mã chương trình Matlab cho kịch bản này được cho bởi file **NVD8_ snrsim.m** (có trong Phụ lục 8A).

Chạy chương trình tạo ra các kết quả sau:

Giá trị của P_x là 3200

Giá trị của P_y là 208,7872

Giá trị độ lợi là 0,25012

Giá trị trễ là 64

Giá trị SNR là 13,6728 dB

Các giá trị lý thuyết dễ dàng được tính toán. Vì tín hiệu tham chiếu là hình sin có giá trị đỉnh là 80 nên:

$$P_x = \frac{1}{2}(80)^2 = 3200 \tag{8.38}$$

Từ (8.37) ta có ba thành phần ở đầu vào máy thu là: tín hiệu sin tại 2 Hz, nhiễu hình sin tại 8 Hz, tạp âm trắng. Công suất P_y là tổng của 3 thành phần:

$$P_y = \frac{1}{2}(20)^2 + \frac{1}{2}(4)^2 + (0{,}8)^2 = 208{,}64 \tag{8.39}$$

Từ (8.36) và (8.37) ta có độ lợi là:

$$G = \frac{20}{80} = 0{,}25 \tag{8.40}$$

Lưu ý rằng, thành phần tín hiệu có chu kỳ 512 mẫu (*x(t)* trải qua 2 chu kỳ là 1024 mẫu), và trễ pha $\pi/4$, độ trễ là:

$$\tau = \frac{\pi/4}{2\pi}(512) = 64 \text{ mẫu} \tag{8.41}$$

SNR là tỷ số của công suất nhiễu cộng với công suất tạp âm trên tín hiệu tại đầu vào máy thu. Đây là tỷ số của số hạng đầu trong (8.39) với tổng của 2 số hạng sau (lưu ý trong trường hợp này nhiễu chính là tạp âm vì nhiễu trực giao với thành phần tín hiệu). Ta có:

$$\frac{S}{N} = \frac{200}{8,64} = 23,1481 = 13,6452 \quad dB \tag{8.42}$$

Các kết quả này được tóm tắt trong bảng 8.2. Các lỗi nhỏ là do phương sai tạp âm σ_n^2 là biến ngẫu nhiên vì chiều dài bản ghi là hữu hạn. Nâng cấp chương trình sao cho 5 ước tính của SNR (đơn vị dB) là các kết quả được tạo ra ở dạng vectơ:

$$\begin{bmatrix}13,6572 & 13,6524 & 13,5016 & 13,5245 & 13,5201\end{bmatrix}$$

Thấy rõ, SNR được ước tính là một biến ngẫu nhiên.

Bảng 8.2: Tóm tắt các kết quả cho ví dụ 8.2

Tham số	Giá trị lý thuyết	Giá trị ước tính
P_x	3.200	3.200
P_y	208,64	208,7872
G	0,25	0,25012
τ	64 mẫu	64 mẫu
S/N	13,6452 dB	13,6728 dB

Khó khăn lớn nhất đối với phương pháp này là xác định chính xác trễ. Lưu ý rằng, nếu xảy ra lỗi nhỏ trong ước tính trễ, thì cũng gây ra lỗi nhỏ trong giá trị được ước tính của $R_{xy}^2(\tau_m)$ nhỏ. Nếu SNR là lớn, thì sẽ dẫn đến kết quả là $P_xP_y - R_{xy}^2(\tau_m) \underset{\approx}{\leq} 0$ và, ta thấy từ (8.30) chỉ một lỗi nhỏ trong ước tính của $R_{xy}^2(\tau_m)$ sẽ gây ra lỗi lớn trong SNR được ước tính. Theo đó, ta phải xác định chính xác giá trị đỉnh của $R_{xy}^2(\tau)$. Điều này cần phải lấy mẫu $R_{xy}^2(\tau)$ nhanh hơn, hay tần số lấy mẫu cao khi mô phỏng. Vì vậy, ta phải dung hoà mọi điểm trong mô hình mô phỏng giữa độ chính xác và thời gian mô phỏng.

Trong quá trình triển khai, ta đã giả định rằng các tín hiệu là các tín hiệu giá trị thực. Kỹ thuật được trình bày ở đây có thể áp dụng cho đường bao phức. Bộ ước tính cho trường hợp này được rút ra bằng cách thay $x(t)$ bằng $x_d(t)+jx_q(t)$ và $y(t)$ bằng $y_d(t)+jy_q(t)$. Biểu thức kết quả đối với SNR cũng là (8.32).

8.4. Mã hoá

Nếu mô phỏng được dùng để xác định BER trong trường hợp hệ thống có sử dụng mã hóa kiểm soát lỗi, *thì* thường không đếm số lỗi tại đầu ra *bộ giải mã*. Một số lý do cơ bản sau: *(i)* BER tại đầu ra bộ giải mã thường rất nhỏ; *(ii)* thời gian chạy mô phỏng rất lâu mới tập hợp số lỗi đủ lớn để các ước tính BER được chính xác; *(iii)* tính phức tạp của nhiều thuật toán giải mã làm tăng đáng kể thời gian chạy mô phỏng; *(iv)* cả bộ mã hoá và giải mã đều là thiết bị tất định. Một khi mã được xác định thì dữ liệu nguồn xác định duy nhất các từ mã đầu ra bộ mã hóa. Tương tự mẫu lỗi đầu ra máy thu xác định duy nhất BER đầu ra bộ giải mã. Điều này gợi

ra giải pháp bán giải tích, trong đó tỉ số lỗi ký hiệu SER tại đầu vào máy thu được xác định bằng mô phỏng, được chuyển thành BER bằng phương pháp phân tích. Ở dạng tổng quát, việc ánh xạ này là một nhiệm vụ phức tạp nếu muốn có các kết quả chính xác. May thay, các kết quả chính xác là hiếm khi cần đến, và một số xấp xỉ và giới hạn đã được triển khai để đơn giản hóa nhiệm vụ này.

Mô phỏng mức dạng sóng thường được sử dụng để xác định tỷ số lỗi ký hiệu SER tại đầu vào máy thu. Một phương pháp thay thế là sử dụng các mô hình kênh rời rạc như các mô hình Markov HMM. HMM là kỹ thuật tính toán hiệu quả để mô phỏng các hệ thống khi cho trước tập các điều kiện kênh, vì vậy là rất hữu hiệu để nghiên cứu ảnh hưởng của các thuật toán mã hoá/giải mã khác nhau. Mô hình kênh rời rạc và HMM sẽ được nghiên cứu chi tiết ở chương 15.

Lưu ý rằng, để được rõ hơn về hoạt động của mã hóa kênh kiểm soát lỗi bạn đọc có thể tham khảo phần phụ lục 8B1. Ở đó trình bày khá chi tiết và dễ hiểu cho quá trình mã hóa và giải mã cũng như hiệu năng của mã, đặc biệt có ý nghĩa đối với sinh viên.

8.4.1. Giải pháp giải tích cho mã khối

Ta biết rằng, các mã khối được tạo ra bằng cách nhóm các ký hiệu thông tin thành các khối có độ dài k. Với mỗi khối k ký hiệu được gắn vào $(n-k)$ ký hiệu kiểm tra, tạo từ mã dài n. Những từ mã này được truyền qua kênh, dưới tác động của kênh gây ra các lỗi ngẫu nhiên. Trong hầu hết các ứng dụng thực tế, các từ mã n ký hiệu được truyền trong một khe thời gian lâu kT_b, trong đó T_b là thời gian để truyền một bit thông tin không mã hoá. Nếu công suất phát là như nhau khi có và không có mã hoá, thì năng lượng tương ứng để truyền các ký hiệu mã là $(k/n)E_b$, trong đó E_b là năng lượng trên bit và k/n là tỷ lệ mã. Vì năng lượng trên ký hiệu phát bị giảm do sử dụng mã hóa kiểm soát lỗi, nên xác suất lỗi ký hiệu kênh có mã hoá bị tăng so với xác suất lỗi ký hiệu (bit) không mã hoá. Hy vọng rằng, thông qua việc thêm các ký hiệu kiểm tra tạo khả năng sửa lỗi hiệu quả để tăng hiệu năng hệ thống. Điều này có thể đúng hoặc không đúng.

Giả sử: *(i)* một mã có thể sửa t lỗi trong mỗi khối n ký hiệu; *(ii)* các sự kiện lỗi là độc lập (có thể được đảm bảo bằng cách sử dụng kỹ thuật đan xen). Xác suất lỗi liên quan với các ký hiệu mã được phát qua kênh ký hiệu là P_{sc}, trong đó chỉ số dưới thể hiện các ký hiệu kênh. Vì bộ giải mã có thể sửa được t lỗi trong một từ mã n ký hiệu nên xác suất mà từ mã được giải mã sẽ trong trạng thái lỗi P_{cw} là:

$$P_{cw} \le \sum_{i=t+1}^{n} \binom{n}{i} P_{sc}^{i} \left(1-P_{sc}\right)^{n-i} \tag{8.43}$$

Dấu bằng xảy ra trong (8.43) nếu tất cả các khối n ký hiệu thu chứa t lỗi hoặc ít hơn t lỗi được giải mã chính xác và không có khối n ký hiệu nào chứa $t+1$ hoặc nhiều hơn $t+1$ lỗi được giải mã chính xác, khi này được biết như là các mã hoàn hảo. Chỉ các mã nhị phân hoàn hảo là các mã lặp đối với n là lẻ, các mã Hamming sửa đơn lỗi và mã Golay (23,12) sửa ba lỗi. Đối với tất cả các mã khác, biểu thức (8.43) cung cấp một giới hạn hữu hiệu.

Xác suất lỗi từ mã được giải mã P_{cw} không cho phép trực tiếp so sánh các mã khác nhau. Để so sánh các mã khác nhau ta cần ánh xạ xác suất lỗi từ mã được giải mã thành xác suất lỗi bit thông tin được giải mã, ký hiệu là P_b. Một ánh xạ chính xác là một hàm ma trận tạo mã của mã đó mà nó xác định phân bố trọng số mã. Nhiều công trình đã triển khai phép xấp xỉ chính xác cao. Phép xấp xỉ này là:

$$P_b \approx \frac{q}{2(q-1)}\left[\frac{d}{n}\sum_{i=t+1}^{d}\binom{n}{i}.P_{sc}^{i}.(1-P_{sc})^{n-i}+\frac{1}{n}\sum_{i=d+1}^{n}\binom{n}{i}.P_{sc}^{i}.(1-P_{sc})^{n-i}\right] \quad (8.44)$$

Trong đó q ký hiệu kênh q tia *(q-ary)*. Nói cách khác, với các kênh nhị phân thì $q = 2$, và với các mã Reed-Solomon mã khối phi nhị phân phổ biến nhất $q = 2^k - 1$.

Ví dụ 8.6: Ta minh hoạ việc sử dụng (8.44). Giả sử hệ thống truyền thông **BPSK** ($q = 2$) hoạt động trong môi trường kênh AWGN. Khi này, xác suất lỗi bit không mã hoá là:

$$P_b = Q\left(\sqrt{2z}\right) = Q\left(\sqrt{2\frac{E_b}{N_0}}\right) \quad (8.45)$$

Trong đó z biểu diễn cho E_b/N_0. Với mã khối *(n,k)*, xác suất lỗi ký hiệu kênh là:

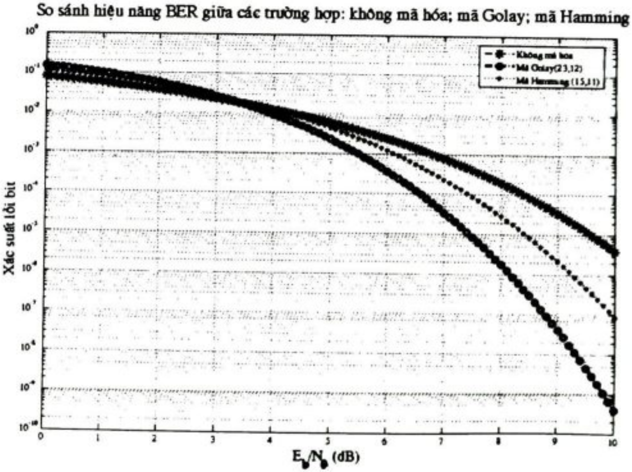
$$P_{sc} = Q\left(\sqrt{\frac{k}{n}2z}\right) = Q\left(\sqrt{\frac{k}{n}\times 2\frac{E_b}{N_0}}\right) \quad (8.46)$$

Xét hai mã nhị phân khác nhau là: mã Golay (23,12) trong đó $n = 23$, $t = 3$ và $d = 7$, và mã Hamming (15,11) trong đó $n = 15$, $t = 1$ và $d = 3$.

Tại điểm này tất cả các tham số và các biến số trong (8.44) đều được biết. Tuy nhiên trước khi ước lượng (8.44) ta phải ước lượng:

$$\binom{n}{i} = \frac{n!}{i!(n-i)!} \quad (8.47)$$

Chương trình Matlab thực hiện (8.47) được cho ở file **NVD8_ nkchoose** (có trong Phụ lục 8A).



Hình 8.13: So sánh hiệu năng các mã khối Golay và Hamming

Chương trình Matlab để tính toán các đường cong hiệu năng cho mã Hamming (15,11) và mã Golay (23,12) sửa ba lỗi được cho ở file **NVD8_ cersim** (có trong Phụ lục 8A). Cần lưu ý rằng, ta đã giả định kênh AWGN và điều chế PSK.

Cần lưu ý rằng, trong chương trình **NVD8_ cersim.m** ta đã dùng chương trình con **NVD8_cer2ber** dưới đây để chuyển xác suất lỗi ký hiệu kênh thành xấp xỉ xác suất lỗi bit giải mã được cho bởi (8.44).

Kết quả của các tính toán này được minh họa ở hình 8.13

8.4.2. Giải pháp giải tích cho mã xoắn

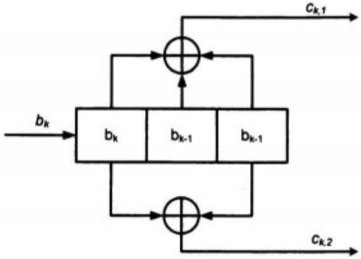
Có thể dùng một số xấp xỉ giải tích để ánh xạ xác suất lỗi ký hiệu kênh thành xác suất lỗi bit được giải mã đối với trường hợp mã xoắn. Những ánh xạ này lấy giới hạn trên của xác suất lỗi và vì vậy tương đương mã xoắn (8.44). Những giới hạn này thường dựa vào thuật toán giải mã Viterbi, gần như tiệm cận với hiệu năng bộ giải mã khả năng giống nhất ML và là tiêu chuẩn để giải mã xoắn.

Giới hạn được sử dụng thường dựa vào hàm truyền đạt của mã xoắn. Hàm truyền đạt mô tả các tính chất khoảng cách của mã xoắn và có thể được rút ra từ biểu đồ chuyển dịch trạng thái của mã. Hàm truyền đạt của bộ lập mã xoắn tỷ lệ 1/2 hình 8.14 được cho bởi:

$$T(D,L,I)=\frac{D^5.L^3.I}{1-D.L.(1+L).I} \quad (8.48)$$

Biểu diễn (8.48) ở dạng đa thức là:

$$T(D,L,I)=\sum_{k=0}^{\infty}D^{5+k}.L^{3+k}.(1+L)^k.I^{1+k}=D^5.L^3.I+D^6.L^4.(1+L)I^2+\dots \quad (8.49)$$



Hình 8.14: Bộ lập mã xoắn tỷ lệ 1/2 cho ví dụ 8.6

Mô tả các tính chất khoảng cách của các đường dẫn khác nhau trong biểu đồ lưới của mã đó mà bắt đầu tại trạng thái toàn 0 và hợp nhất lại trạng thái toàn 0 sau đó. Số mũ của D thể hiện khoảng cách Hamming (số các số 1 nhị phân) phân cách đường dẫn đã cho so với đường dẫn toàn 0 trong lưới giải mã. Số mũ của L cho biết chiều dài của một đường dẫn đã cho. Nói cách khác, luỹ thừa của L được tăng lên mỗi khi đi qua một nhánh trong lưới. Số mũ của I được tăng lên nếu sự chuyển dịch nhánh do đầu vào bit 1 nhị phân, và không được tăng nếu sự

chuyển dịch nhánh là kết quả từ đầu vào là bit 0. Ví dụ, số hạng D^5L^3I biểu diễn cho một đường dẫn có khoảng cách Hamming là 5 so với đường dẫn toàn 0. Đường dẫn này có độ dài là 3 và kết quả từ dữ liệu đầu vào có một bit nhị phân bằng 1 và hai bit nhị phân 0 (tức là 100). Số hạng tiếp theo, $D^6L^4(1+L)I^2 = D^6L^4I^2+D^6L^5I^2$, biểu diễn cho 2 đường dẫn, mỗi đường có khoảng cách Hamming là 6 so với đường dẫn toàn 0. Một đường có chiều dài 4 nhánh và đường kia có chiều dài 5 nhánh. Đường có chiều dài 4 nhánh là kết quả từ đầu vào 2 bit 1 và 2 bit 0, và đường kia là kết quả từ đầu vào có 2 bit 1 và 3 bit 0. Trọng số đầu ra nhỏ nhất của tất cả các đường dẫn bắt đầu và hợp nhất với trạng thái tất cả bit 0 thể hiện khoảng cách tự do nhỏ nhất d_f của mã, trong trường hợp này là 5.

Với mục đích xấp xỉ hóa xác suất lỗi bit giải mã. Trước hết, ta đặt $L = 1$ trong (8.48), vì ta không quan tâm đến chiều dài đường dẫn. Nên có:

$$T(D,I) = T(D,L,I)\big|_{L=1} = \frac{D^5.I}{1-2D.I} \tag{8.50}$$

Đối với tín hiệu PSK đối cực trong môi trường AWGN, thì xác suất lỗi ký hiệu giải mã được cho bởi:

$$P_E < R\frac{\partial T(D,I)}{\partial I}\bigg|_{I=1,D=e^{RE_b/N_0}} \tag{8.51}$$

Trong đó R là tỷ lệ mã (trường hợp này $R = 1/2$). Kết quả này được sử dụng trong ví dụ 8.6. Đối với kênh đối xứng nhị phân tổng quát, thì giới hạn Bhattachayya được dùng. Điều này dẫn đến:

$$P_E = \frac{\partial T(D,I)}{\partial I}\bigg|_{I=1,D=d} \tag{8.52}$$

Trong đó:

$$d = \sqrt{4q(1-q)} \tag{8.53}$$

Và q là xác suất lỗi ký hiệu kênh được xác định bởi mô phỏng. Các giới hạn được định nghĩa bởi (8.51) và (8.52) có thể không khá chặt chẽ. Điều này đặc biệt đúng với các mã chiều có dài hạn chế ngắn.

Các phương trình (8.51) và (8.52) được giả định cho giải mã quyết định cứng. Với giải mã quyết định mềm, thì d trong (8.51) được thay bằng d_0.

Trong đó:

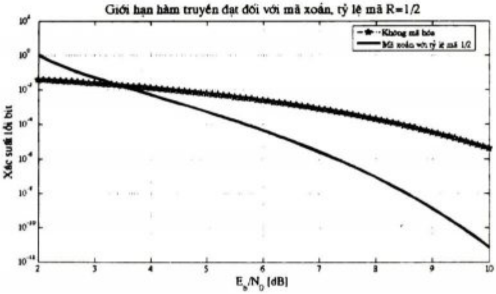
$$d_0 = \sum_{i=1}^{N} \sqrt{\Pr(y_i \mid 0)\Pr(y_i \mid 1)} \tag{8.54}$$

N là số các giá trị đầu ra bộ lượng tử hoá, y_i là giá trị đầu ra được lượng tử hoá thứ i, và các xác suất có điều kiện biểu diễn cho xác suất của các bit 0 và 1 tại đầu vào kênh sẽ xuất hiện tại đầu ra bộ lượng tử hoá là mức y_i. Các xác suất có điều kiện này được ước tính bằng kỹ thuật Monte Carlo hoặc bán phân tích trên kênh dạng sóng.

Ví dụ 8.7: Ta áp dụng (8.51) cho mã xoắn có tỷ lệ mã là 1/2 được định nghĩa bởi (8.50) và được minh hoạ trong hình 8.14. Thay (8.50) vào (8.52) tạo ra:

$$P_E < \frac{1}{2}\frac{\partial}{\partial I}\left(\frac{D^5 I}{1-2DI}\right)\Bigg|_{I=1, D=e^{-RE_b/N_0}} = \frac{1}{2}\frac{D^5}{(1-2D)^2}\Big|_{D=e^{-RE_b/N_0}} \tag{8.55}$$

Mã chương trình Matlab thực hiện bài toán này được cho ở file **NVD8_convcode.m** (có trong Phụ lục 8A). Kết quả chạy chương trình Matlab được cho ở hình 8.15 trong đó so sánh giới hạn xác suất lỗi giải mã và xác suất lỗi không được giải mã.

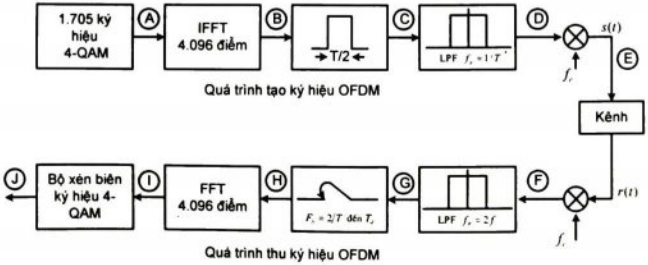


Hình 8.15: Giới hạn hàm truyền đạt đối với ví dụ mã xoắn tỷ lệ mã 1/2

8.5. Một số ứng dụng điển hình

Chương trình 1: Mô phỏng tín hiệu và hệ thống OFDM

Chương trình mô phỏng tín hiệu OFDM được cho ở file **NVD8_ofdmsimtx.m** và **NVD8_ofdmsimrx.m** trong Phụ lục 8A, mô phỏng tín hiệu OFDM trong miền thời gian và tần số. Mặc dù mục đích ở đây là trình bày cách dùng các thường trình hậu xử lý để trực quan hóa tín hiệu và hệ thống trong mô phỏng, nhưng chương trình này rất hữu hiệu để tìm hiểu cách thiết kế và thực hiện mô phỏng hệ thống OFDM đơn giản. Từ chương trình này cho phép nâng cấp để tạo chương trình mô phỏng hệ thống phức tạp hơn. Trình bày chi tiết về việc thiết kế và mô phỏng tín hiệu OFDM được cho ở Phụ lục **8B2**. Để tiện theo dõi, sơ đồ mô phỏng quá trình phát thu tín hiệu OFDM được minh họa ở hình 8.16.



Hình 8.16: Sơ đồ mô phỏng quá trình tạo và thu tín hiệu OFDM

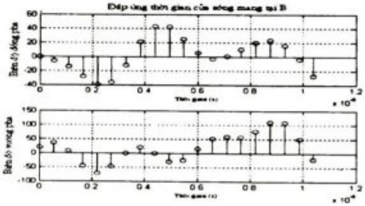
Ta mô phỏng tín hiệu OFDM có cấu trúc khung và đa khung. Theo đó, mỗi khung có độ lâu T_F gồm N_F ký hiệu OFDM, mỗi đa khung gồm M_F khung ($T_{MF} = M_F \times T_F$). Mỗi ký hiệu được tạo thành bởi một tập K (từ K_{min} đến K_{max}) sóng mang và được truyền trong khoảng thời gian T_S; $T_S = T_U + \Delta$. Các thông số cụ thể được cho ở bảng 8.3. Cũng cần lưu ý rằng, để được tường minh ta nên đọc kỹ Phụ lục 8B2 và các chương trình mô phỏng. Ở đó trình bày khá rõ về quá trình thiết kế và mô phỏng cho trường hợp đơn giản điển hình này. Các kết quả mô phỏng tương ứng với các điểm được đánh nhãn trên sơ đồ mô phỏng 8.16.

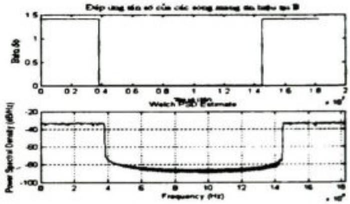
Bảng 8.3: Các giá trị đối với các thông số OFDM trong mô phỏng

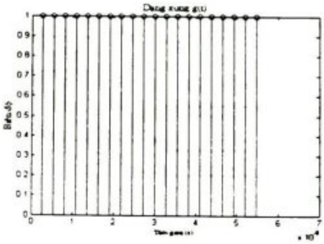
Thông số	Giá trị			
Chu kỳ cơ bản T	7/64 μs			
Số sóng mang K (trong khoảng K_{min} đến K_{max})	1705			
Giá trị số sóng mang K_{max}	1704			
Giá trị số sóng mang K_{min}	0			
Khoảng thời gian ký hiệu hữu hiệu T_U	224 μs			
Khoảng cách sóng mang $1/T_U$	4464 Hz			
KHoảng cách giữa các sóng mang K_{min} và $K_{max}(K - 1)T_U$	7,61 MHz			
Khoảng thời gian bảo vệ được phép Δ/T_U	1/4	1/8	1/16	1/32
Khoảng thời gian phần T_U	2048 × T = 224 μs			
Khoảng thời gian bảo vệ Δ	512 × T = 56 μs	256 × T = 28 μs	128 × T = 14 μs	64 × T = 7 μs
Khoảng thời gian ký hiệu $T_S = \Delta + T_U$	2.560 × T = 280 μs	2.304 × T = 252 μs	1.176 × T = 238 μs	2.112 × T = 231 μs

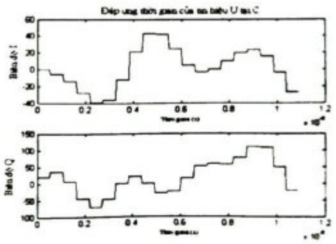
Sử dụng các chương trình thường trình hậu xử lý để trình bày các tín hiệu tại các điểm khác nhau trong sơ mô phỏng như sau:

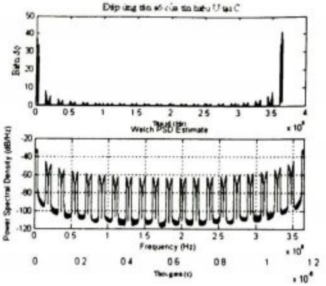
Kết quả mô phỏng phần phát

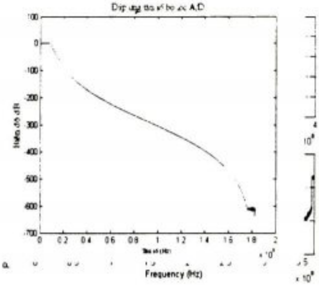


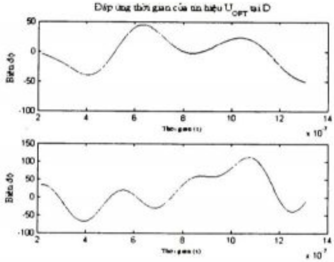


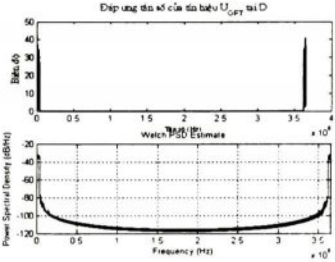


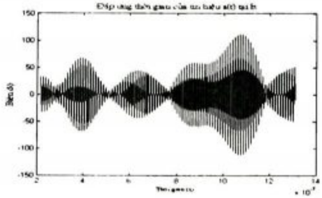


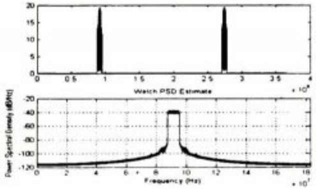




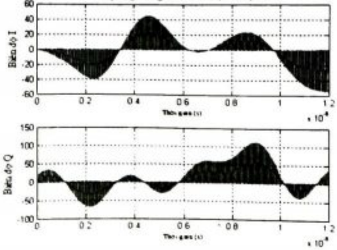


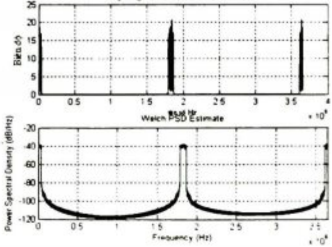


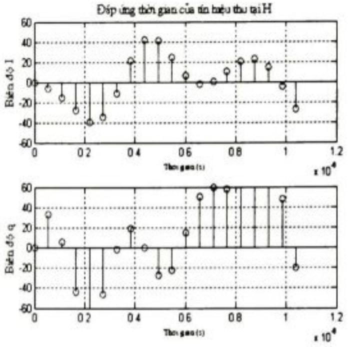


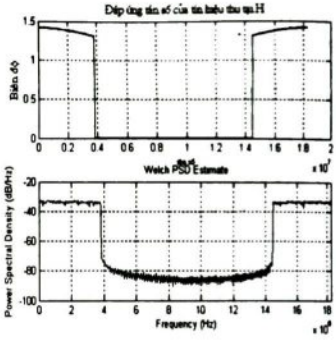


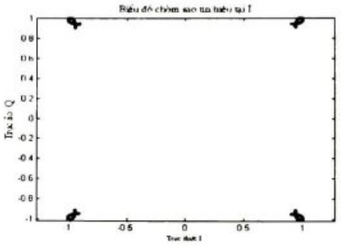
Kết quả mô phỏng phần thu

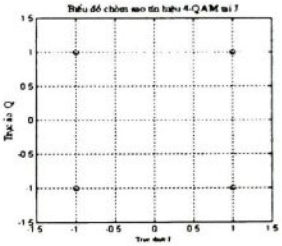








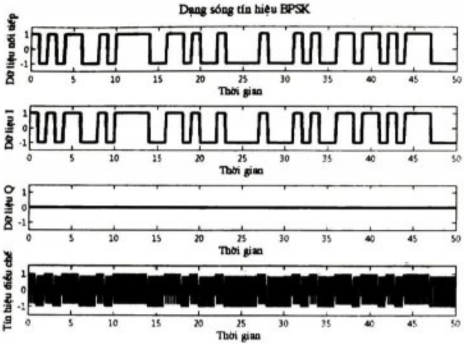


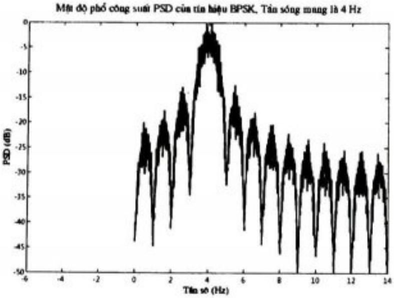


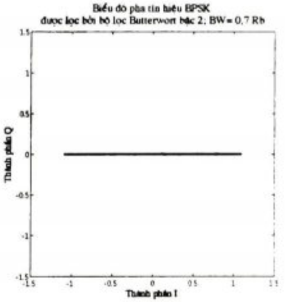
Chương trình 2: Mô phỏng tín hiệu và hệ thống BPSK, QPSK, OQPSK, MSK, 16QAM

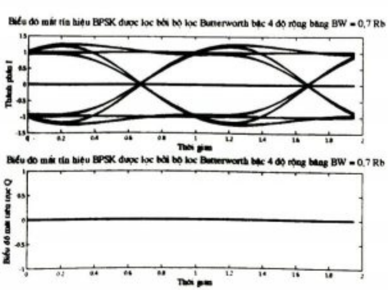
Chương trình mô phỏng tín hiệu và hệ thống BPSK, QPSK, OQPSK, MSK, 16QAM được cho ở file **NVD8_sum1.m** trong Phụ lục 8A thực hiện mô phỏng tín hiệu trong miền thời gian và tần số như: dạng sóng, biểu đồ mắt , biểu đồ pha (chòm sao), đường bao, mật độ phổ công suất PSD. Dưới đây là một số kết quả mô phỏng điển hình cho các hệ thống này. Để được tường minh, ta nên nghiên cứu kỹ mã chương trình Matlab, xác định các tham số đầu vào cho từng hệ thống, vẽ lưu đồ mô phỏng. Đây là một chương trình tổng hợp có tính liên kết cao. Khi đó, trình bày rõ cách lấy dữ liệu từ các điểm của hệ thống để xử lý hiển thị, trực quan hóa nguyên lý hoạt động của các hệ thống này. Đặc biệt có ý nghĩa đối với sinh viên, người mới lập trình Matlab và muốn tìm hiểu nguyên lý hoạt động các hệ thống truyền thông này. Dưới đây là một số kết quả điển hình khi chạy chương trình mô phỏng. Nếu mục đích là hiểu nguyên lý hoạt động các hệ thống này, thì nên dành thời gian nghiên cứu kỹ mã chương trình Matlab này, vẽ lại mô hình mô phỏng, công thức hóa cho từng đoạn chương trình và chạy chương trình mô phỏng cho tập các tham số đặc trưng khác nhau cho từng mô hình đó. Có thể nói đây là một bài tập rất tốt.

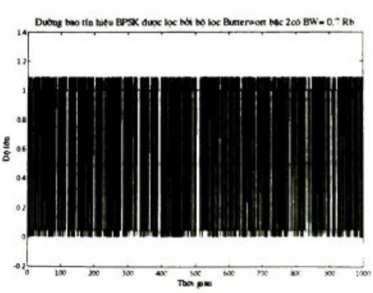
Hệ thống BPSK:



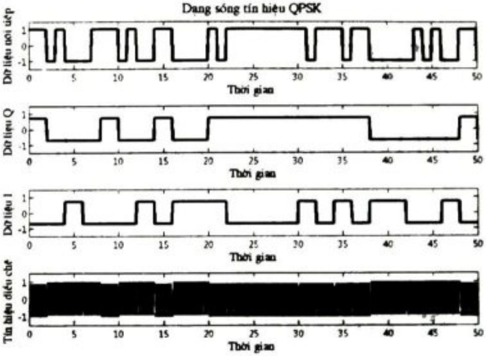


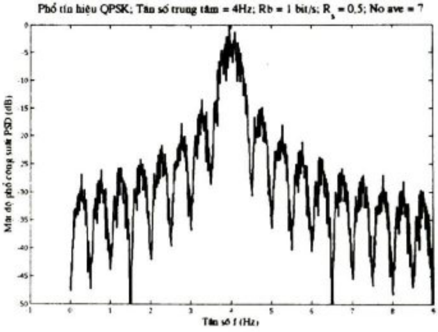


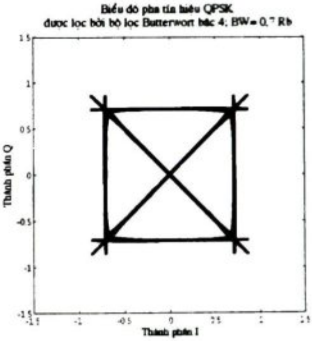


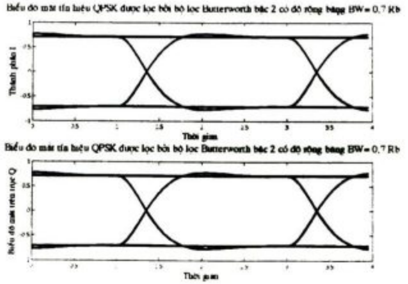


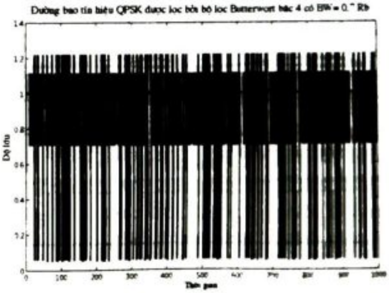
Hệ thống QPSK



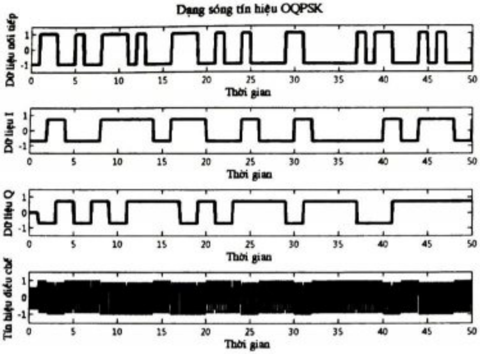


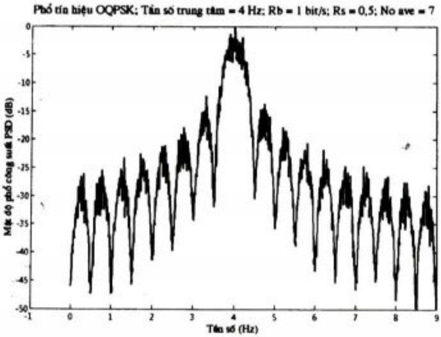


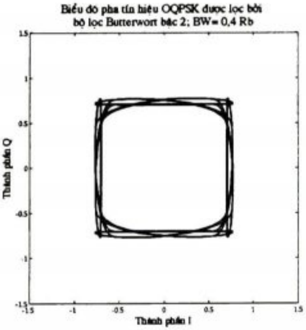


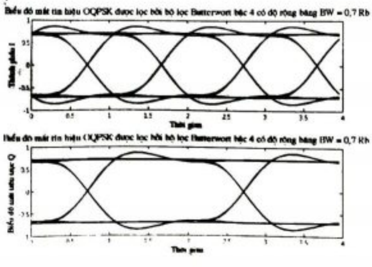


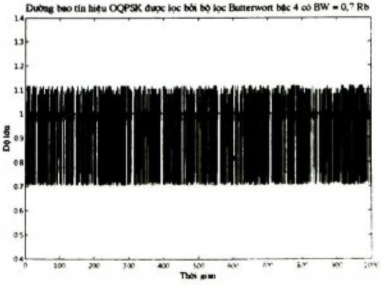
Hệ thống OQPSK



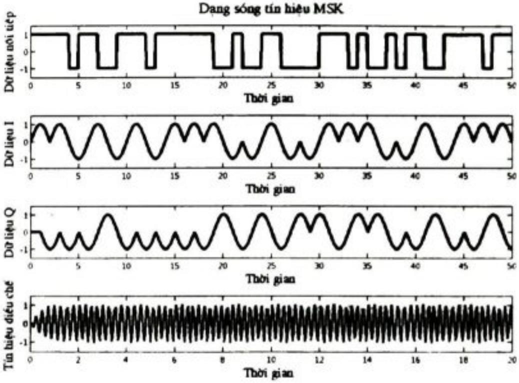


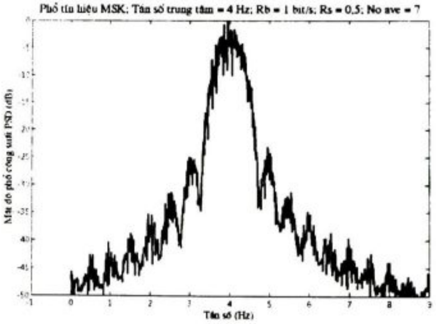


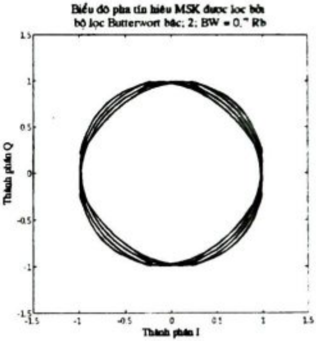


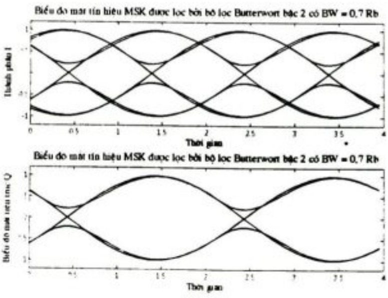


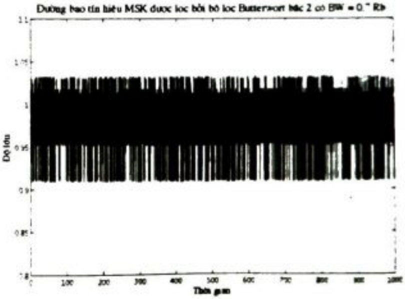
Hệ thống MSK


Dạng sóng tín hiệu MSK
Dữ liệu nối tiếp
Dữ liệu I
Dữ liệu Q
Tín hiệu điều chế
Thời gian

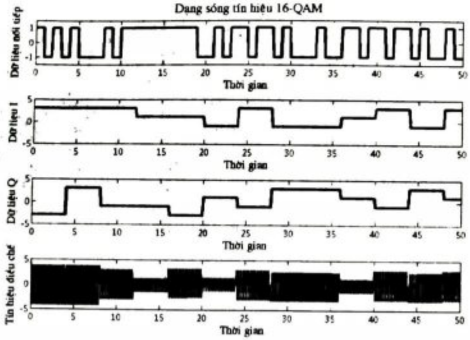

Phổ tín hiệu MSK; Tần số trung tâm = 4 Hz; Rb = 1 bit/s; Rs = 0,5; No ave = 7
Mật độ phổ công suất PSD (dB)
Tần số (Hz)

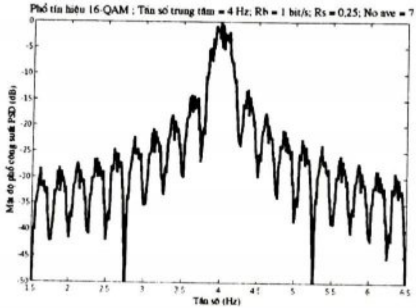

Biểu đồ pha tín hiệu MSK được lọc bởi bộ lọc Butterwort bậc; 2; BW = 0.7 Rb
Thành phần Q
Thành phần I

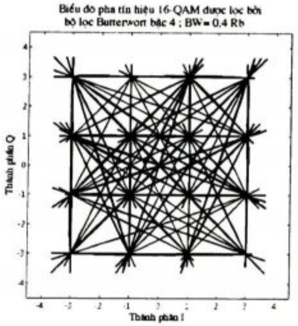

Biểu đồ mắt tín hiệu MSK được lọc bởi bộ lọc Butterwort bậc 2 có BW = 0,7 Rb
Thời gian
Biểu đồ mắt tín hiệu MSK được lọc bởi bộ lọc Butterwort bậc 2 có BW = 0,7 Rb
Thời gian

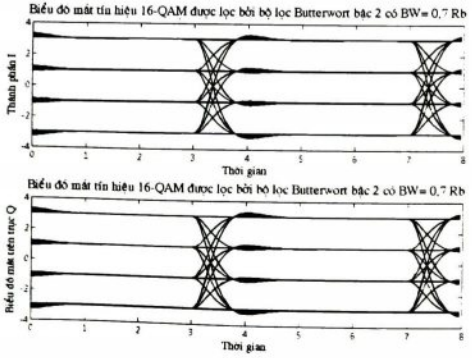

Đường bao tín hiệu MSK được lọc bởi bộ lọc Butterwort bậc 2 có BW = 0.7 Rb
Độ lớn
Thời gian

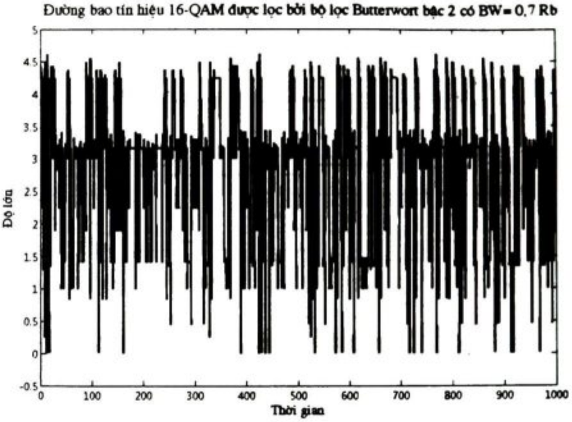
Hệ thống 16-QAM










Đường bao tín hiệu 16-QAM được lọc bởi bộ lọc Butterwort bậc 2 có BW= 0,7 Rb
Độ lớn
Thời gian
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
0
100
200
300
400
500
600
700
800
900
1000

Chương 9
GIỚI THIỆU PHƯƠNG PHÁP MONTE CARLO

9.1. Mở đầu

Mục đích của chương là giới thiệu vắn tắt các cơ sở của kỹ thuật Monte Carlo để ước tính giá trị của tham số. Vì thế, ta chỉ đề cập vài khía cạnh quan trọng của kỹ thuật ước tính Monte Carlo nhằm định nghĩa phương pháp Monte Carlo và nghiên cứu một số kỹ thuật căn bản một cách đơn giản và dễ hiểu. Ta cũng đề cập ngắn gọn các vấn đề quan trọng về các khoảng tin cậy, tính hội tụ. Trong toàn bộ chương ta đều coi rằng, các quan trắc bởi bộ ước tính là độc lập. Giả định này sẽ được giảm nhẹ trong các chương sau ở đó xét các kỹ thuật mô phỏng chi tiết hơn.

Muốn vậy, chương sẽ đề cập mô phỏng và ước tính Monte Carlo sử dụng một số minh họa đơn giản. Ta sẽ thấy rằng, các kỹ thuật Monte Carlo dựa trên hiệu năng của những thí nghiệm ngẫu nhiên (stochatic). Nhận biết sự kiện nghiên cứu và thí nghiệm ngẫu nhiên cơ bản được tái tạo nhiều lần. Tỉ số giữa số lần xuất hiện sự kiện nghiên cứu với tổng số lần tái tạo thí nghiệm ngẫu nhiên cho ta tần xuất tương đối của sự kiện nghiên cứu. Tần xuất tương đối là một biến ngẫu nhiên và là bộ ước tính xác suất của sự kiện nghiên cứu. Đối với các bộ ước tính *kiên định* và *không chệch*, thì tần xuất tương đối sẽ hội tụ đến xác suất của sự kiện đang xét khi số lần tái tạo lớn.

Các kết quả đạt được của chương minh họa cho việc phân biệt rất quan trọng giữa mô phỏng stochastic và phân tích toán học truyền thống. Khi dùng phân tích toán học truyền thống để xác định giá trị của thông số, thì kết quả đó tuyệt đại đa số là một con số. Ví dụ khi phân tích một hệ thống truyền thông số cho ta kết quả P_E =1,7638-3 tất nhiên nó chỉ là một con số. Tuy nhiên, khi dùng các kỹ thuật Monte Carlo cho ta kết quả là một biến ngẫu nhiên. Các thuộc tính của biến ngẫu nhiên này như: trung bình, phương sai, hàm mật độ xác suất..... cho ta nhiều thông tin về chất lượng của kết quả mô phỏng.

9.2. Khái niệm cơ bản

Các mô phỏng Monte-Carlo dựa trên những trò chơi may rủi. Tất nhiên, là lý do của tên "Monte Carlo", thành phố thuộc Địa trung hải nổi tiếng về các casino. Ta sử dụng hai thuật ngữ quan hệ mật thiết "*ước tính Monte Carlo*" và "*mô phỏng Monte Carlo*", có thể thay thế cho nhau. Mô phỏng Monte Carlo mô tả mô phỏng trong đó một tham số của hệ thống như tỷ số lỗi bit (BER) được ước tính bằng cách sử dụng các kỹ thuật Monte Carlo. Ước tính Monte Carlo là quá trình ước tính giá trị của một tham số bằng cách thực hiện quá trình stochastic cơ bản hay thí nghiệm ngẫu nhiên.

9.2.1. Tần suất tương đối

Ước tính Monte Carlo dựa trên biểu diễn tần suất tương đối của xác suất. Trong quá trình xác định tần suất tương đối ta thực hiện như sau:

Trước hết cần phải xác định rõ thí nghiệm ngẫu nhiên và một sự kiện quan tâm *A*. Thấy rõ từ lý thuyết xác xuất cơ bản, một thí nghiệm ngẫu nhiên là một thí nghiệm trong đó kết quả (hoặc kết cục) của việc thực hiện thí nghiệm không thể dự đoán chính xác được *nhưng* có thể được xác định một cách thống kê. Thí nghiệm ngẫu nhiên cơ bản nhất là tung đồng xu trong đó có hai kết cục quan tâm được định nghĩa bởi tập *{xấp, ngửa}*. Trước khi tung đồng xu, ta hoàn toàn không thể biết trước được kết cục nào sẽ xẩy ra. Tuy nhiên, nếu biết đồng xu "*trung thực*" (không thiên vị), thì xác suất của mỗi kết cục trong tập *{xấp, ngửa}* sẽ xẩy ra với cùng khả năng và các kết cục đó là độc lập nhau. Hiệu năng của thí nghiệm ngẫu nhiên đó xác định kết cục.

Một sự kiện là một hoặc một tập các kết cục liên quan với một thí nghiệm ngẫu nhiên. Chẳng hạn, trong hệ thống viễn thông số, thí nghiệm ngẫu nhiên có thể chỉ là việc phát bit 1 nhị phân. Kết quả tại đầu ra của máy thu là một ước tính về bit nhị phân đã được phát đi đó (là 0 hoặc 1). Trong quá trình truyền dẫn bit 1 sự kiện quan tâm có thể xẩy ra lỗi. Việc xác định hiệu năng BER của hệ thống bao gồm ước tính xác suất có điều kiện là xác suất thu được bit 0 biết rằng (với điều kiện) bit 1 đã được phát đi.

Sau đó, một khi đã định nghĩa thí nghiệm ngẫu nhiên và sự kiện quan tâm, tiếp theo là thực hiện thí nghiệm ngẫu nhiên N lần. Đếm số lần xảy ra sự kiện quan tâm A với kết quả là NA.

Sau cùng, xác định xác suất xẩy ra sự kiện quan tâm theo định nghĩa tần suất tương đối của xác suất. Xác suất xảy ra sự kiện *A* được xấp xỉ hóa bởi tần suất xuất hiện tương đối của sự kiện *A* là N_A/N. Xác suất xảy ra sự kiện *A* được định nghĩa theo tần suất tương đối bằng cách tái tạo thí nghiệm ngẫu nhiên với vô hạn lần:

$$\Pr(A) = \lim_{N \to \infty} \frac{N_A}{N} \tag{9.1}$$

Trong ước tính xác suất lỗi ở một hệ thống truyền dẫn số, *N* là tổng số bit hoặc ký hiệu (thực tế đã được phát đi hoặc được mô phỏng) và N_A là số lỗi (được đo hoặc được mô phỏng).

Khi $N<\infty$, hiển nhiên trong mô phỏng Monte Carlo đại lượng N_A/N là một ước tính của *Pr(A)* ký hiệu là $\hat{\Pr}(A)$. Lưu ý rằng, vì thí nghiệm ngẫu nhiên cơ bản được thực hiện *N* hữu hạn lần nên N_A sẽ là một biến số ngẫu nhiên, kết quả ước tính $\hat{\Pr}(A)$ cũng là một biến ngẫu nhiên. Tính thống kê của biến ngẫn nhiên này xác định tính chính xác của bộ ước tính và vì vậy là chất lượng của mô phỏng.

9.2.2. Bộ ước tính không chệch và kiên định

Để hữu hiệu, các bộ ước tính Monte Carlo phải thỏa mãn một số tính chất.

Bộ ước tính không chệch

Ta mong muốn có bộ ước tính Monte Carlo phải *không chệch*, nghĩa là nếu $\hat{A}$ là ước tính của tham số *A*, thì:

$$E\left\{\hat{A}\right\} = A \tag{9.2}$$

Nói cách khác, về phương diện trung bình đạt được kết quả chính xác.

Bộ ước tính kiên định

Giả sử thực hiện một mô phỏng Monte Carlo một số lần và nhận được tập hợp các ước tính của biến ngẫu nhiên quan tâm. Rõ ràng ta mong muốn các ước tính này có phương sai nhỏ. Nếu ước tính là không chệch và có phương sai nhỏ thì bộ ước tính sẽ tạo ra các ước tính tập trung quanh giá trị đúng của tham số được ước tính, và sự trải rộng ước các ước tính đó là nhỏ. Việc xác định phương sai của bộ ước tính Monte Carlo theo phương pháp giải tích là một bài toán khó, trừ khi các sự kiện cơ bản là độc lập thống kê. Tuy nhiên, hầu hết phương sai của các giá trị được ước tính giảm khi tăng thời gian mô phỏng (tăng số lần thực hiện lại thí nghiệm ngẫu nhiên cơ bản đó). Ta coi các bộ ước tính thỏa mãn tính chất này là bộ ước tính *kiên định*. Đối với bộ ước tính kiên định $\sigma_{\hat{A}}^2 \to 0$ khi $N \to \infty$, N là số lần thí nghiệm ngẫu nhiên được tái tạo.

Bộ ước tính không chệch và kiên định

Khi bộ ước tính vừa có tính không chệch vừa có tính kiên định, thì lỗi :

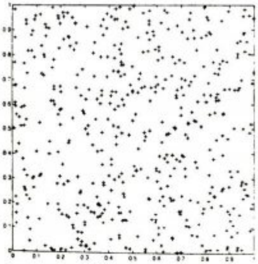
$$e = A - \hat{A} \tag{9.3}$$

Có trung bình không và phương sai σ_e^2 hội tụ về 0 khi $N \to \infty$. Đáng tiếc là sự hội tụ này thường rất chậm.

9.2.3. Ước tính Monte Carlo

Xét ví dụ đơn giản về một bộ ước tính Monte Carlo, xác định diện tích của vùng có hình dạng không bình thường. Giả sử vùng được ước tính hoàn toàn được giới hạn trong diện tích của một hộp đã biết. Định nghĩa thí nghiệm ngẫu nhiên là: lấy ra các mẫu ngẫu nhiên trong hộp giới hạn đó và định nghĩa sự kiện quan tâm A (sự kiện mà một mẫu nằm trong vùng diện tích sẽ được xác định). Vì bộ ước tính vùng không biết đó không chệch, nên các điểm mẫu ngẫu nhiên được phân bố đều trong vùng lấy giới hạn của diện tích đã cho. Điều này được thực hiện một cách dễ dàng bằng chương trình máy tính với hai bộ tạo số ngẫu nhiên đều. Kết quả của việc tạo N = 500 điểm lấy mẫu phân bố đều được minh họa ở hình 9.1. Mã chương trình Matlab dưới đây thực hiện cho vấn đề này:

```
x = rand(1,500);
y = rand(1,500);
plot(x,y,'k+');
axis square
```

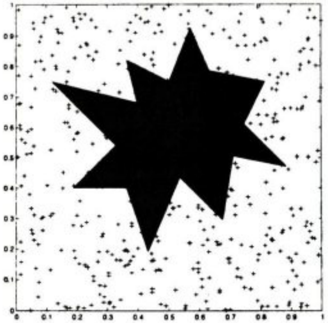


Hình 9.1: Minh họa các điểm ngẫu nhiên phân bố đều

Bước tiếp theo là định nghĩa sự kiện A. Ta mong muốn ước tính diện tích hình sao bôi đen được minh họa ở hình 9.2. Định nghĩa các đại lượng $N_{hộp}$ và N_{sao} là số điểm mẫu nằm trong hộp bao và trong vùng hình sao. Vì các điểm mẫu được phân bố đều trong hộp bao, nên tỉ số giữa diện tích hình sao trên diện tích hộp bao $A_{sao}/A_{hộp}$ là xấp xỉ bằng tỉ số giữa số điểm mẫu nằm trong diện tích hình sao và số điểm mẫu nằm trong diện tích hộp bao $N_{sao}/N_{hộp}$. Nói cách khác ta có:

$$\underbrace{\frac{\text{Diện tích hình sao}}{\text{Diện tích hộp bao}} = \frac{A_{sao}}{A_{hộp}} \approx \frac{N_{sao}}{N_{hộp}} = \frac{\text{Số điểm mẫu trong hình sao}}{\text{Số điểm mẫu trong hình bao}}}_{\substack{\text{Với điều kiện các điểm mẫu được phân bố đều trong hình bao} \\ \text{Phép xấp xỉ càng chính xác nếu số điểm mẫu càng lớn}}} \quad (9.4)$$

$$\Rightarrow A_{sao} \approx A_{hộp}\frac{N_{sao}}{N_{hộp}} \quad (9.5)$$



Hình 9.2: Ước tính Monte Carlo của diện tích

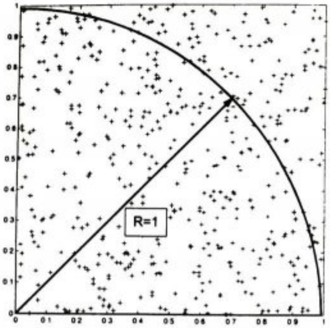
Với điều kiện các điểm mẫu được phân bố đều, phép xấp xỉ sẽ chính xác hơn khi số điểm mẫu được tăng lên.

Để minh họa kĩ thuật Monte Carlo theo cách đơn giản và dễ hiểu, ta xét ước tính Monte Carlo cho giá trị của π. Lưu ý rằng, bộ ước tính là một mô phỏng ngẫu nhiên bởi vì nó là một mô phỏng của thí một nghiệm ngẫu nhiên. Vì vậy ví dụ này tạo ra căn bản để trình bày những chương sau.

9.2.4. Ước tính π

Một phương pháp ước tính giá trị π là bao một diện tích hình dạng tròn, tương ứng với góc phần tư thứ nhất của vòng tròn bán kính đơn vị, bởi một hộp diện tích đơn vị. Điều này được minh họa ở hình 9.3 cùng với toàn bộ Nhộp điểm mẫu. Nếu hộp mở rộng phạm vi (0,1) trên trục x và (0,1) trên trục y, thấy rõ $A_{hộp} = 1$ và diện tích của vùng hình dạng tròn đó (1/4 hình tròn) là:

Giả sử các mẫu được phân bố đồng đều, thì tỉ số $N_{1/4\ hình\ tròn}/N_{hộp}$ sẽ tạo thành một bộ ước tính kiên định và không chệch của $A_{1/4\ hình\ tròn}/A_{hộp}$. Vì vậy:



Hình 9.3: Ước tính π

$$\frac{N_{1/4\text{ hình tròn}}}{N_{\text{hộp}}} \approx \frac{A_{1/4\text{ hình tròn}}}{A_{\text{hộp}}} = \frac{\pi}{4} \qquad (9.8)$$

$$\text{Diện tích 1/4 hình tròn} = A_{1/4\text{ hình tròn}} = \frac{1}{4}\left[\pi R^2\right]_{R=1} = \frac{\pi}{4} \qquad (9.6)$$

$$\Rightarrow \frac{\text{Diện tích 1/4 hình tròn}}{\text{Diện tích hộp bao}} = \frac{A_{1/4\text{ hình tròn}}}{A_{\text{hộp}}} = \pi/4 \qquad (9.7)$$

Ước tính của π ký hiệu $\hat{\pi}$ là:

$$\hat{\pi} = 4\times\frac{\text{Số điểm mẫu trong cung phần tư thứ nhất của hình tròn}}{\text{Số điểm mẫu trong hộp bao}} = 4\times\frac{N_{1/4\text{ hình tròn}}}{N_{\text{hộp}}} \qquad (9.9)$$

Theo đó, có thể ước tính giá trị của π bằng cách che phủ hộp bao bằng những điểm phân bố đồng đều, sau đó đếm số điểm nằm trong vòng tròn nội tiếp và áp dụng (9.9).

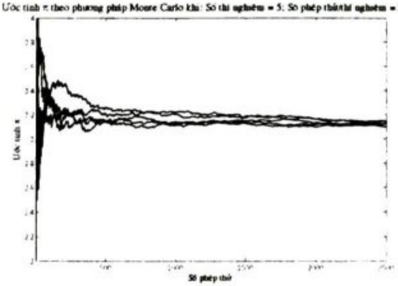
Ví dụ 9.1: Chương trình Matlab thực hiện bài toán này được cho dưới đây. Kết quả được cho ở hình 9.4 tạo ra 5 ước tính khác nhau của π với mỗi ước tính dựa trên 500 bản sao thí nghiệm ngẫu nhiên cơ bản. Năm kết giá trị ước tính của π được xác định nghĩa bởi véc-tơ:

$$\hat{\pi} = [3{,}0960\ \ 3{,}0720\ \ 2{,}9920\ \ 3{,}1600\ \ 3{,}0480] \qquad (9.10)$$

Nếu lấy trung bình 5 ước tính thì kết quả $\hat{\pi}$ = 3,0736. Kết quả này tương đương một ước tính dựa trên 2500 thử nghiệm. Chương trình Matlab để tạo ra những kết quả này được cho bởi **NVD9_estimatepi.m** (có trong Phụ lục 9A)

Trong khi ví dụ trên là rất đơn giản, nó chia sẻ một số thuộc tính quan trọng với tất cả các mô phỏng Monte Carlo. Một điều kiện kiểm tra và một cặp bộ đếm. Bộ đếm đầu tiên được tăng lên mỗi khi thí nghiệm ngẫu nhiên được thực hiện và bộ đếm còn lại được tăng lên mỗi khi điều kiện kiểm tra được thỏa mãn. Trong mô phỏng các hệ thống truyền thông số, với mục đích ước tính BER, thì điều kiện kiểm tra xác định xem xảy ra lỗi trong quá trình truyền một bit hoặc

một ký hiệu tin không. Bộ đếm thứ nhất được tăng lên mỗi khi một bít hoặc một ký hiệu dữ liệu được xử lý bởi mô phỏng. Bộ đếm thứ hai được tăng lên mỗi khi quan trắc được một lỗi. Vấn đề này được sáng tỏ trong cặp ví dụ phần sau. Trước hết ta xét các đặc tính kênh AWGN.



Hình 9.4: Ước tính Monte Carlo của π

9.3. Ứng dụng vào hệ thống truyền thông - Kênh AWGN

Để ước tính hiệu năng hệ thống truyền thông số dùng mô phỏng Monte Carlo, ta cho N ký hiệu qua hệ thống (hình hóa mô phỏng của hệ thống trên máy tính), sau đó đếm số lỗi truyền dẫn N_e. Nếu xảy ra N_e lỗi trong N lần truyền ký hiệu, thì bộ ước tính xác suất lỗi ký hiệu là:

$$\hat{P}_E = \frac{N_e}{N} \tag{9.11}$$

Ước tính này có bị chệch hay không chệch? Ước tính này có kiên định không?

Để trả lời các câu hỏi quan trọng này trong ngữ cảnh đơn giản nhất có thể, ta sẽ coi rằng kênh AWGN (kênh tạp âm Gausơ trắng cộng). Trong môi trường kênh AWGN, các sự kiện lỗi do tạp âm gây ra là độc lập và số lỗi Ne trong truyền dẫn N ký hiệu được mô tả bởi phân bố nhị thức. Vì vậy, ta xét phân bố nhị thức ở mức chi tiết hơn. Sau đó, ta xét biểu thức (9.11) là bộ ước tính xác suất lỗi ký hiệu trong hai hệ thống truyền thông được lý tưởng hóa khá cao.

9.3.1. Phân bố nhị thức

Phần này sẽ xác định tính cách thống kê của ước tính xác suất lỗi ký hiệu $\hat{P}_E$. Muốn vậy,

(1) Trước tiên là xác định trung bình và phương sai của N_e. Vì các sự kiện lỗi độc lập, nên xác suất xảy ra N_e lỗi trong N lần truyền dẫn ký hiệu được cho bởi phân bố nhị thức:

$$p_N(N_e) = \binom{N}{N_e} P_E^{N_e} (1-P_E)^{N-N_e} \tag{9.12}$$

Trong đó:

$$\binom{N}{N_e} = \frac{N!}{N_e!(N-N_e)!} \tag{9.13}$$

Là hệ số nhị thức và P_E là xác suất lỗi trên một lần truyền dẫn.

Dễ dàng rút ra được trung bình và phương sai của biến ngẫu nhiên tuân theo phân bố nhị thức. Trung bình và phương sai của N_e được cho bởi:

$$E[N_e] = N.P_E \quad (9.14)$$

$$\sigma_{N_e}^2 = NP_E(1-P_E) \quad (9.15)$$

(2) Sau đó, xác định trung bình và phương sai của bộ ước tính xác suất lỗi ký hiệu $\hat{P}_E$. Sử dụng kết quả (9.11), trung bình của bộ ước tính Monte Carlo cho xác suất lỗi là:

$$E\{\hat{P}_E\} = \frac{E\{N_e\}}{N} \quad (9.16)$$

Thay (9.14) vào (9.16) ta có:

$$E\{\hat{P}_E\} = \frac{NP_e}{N} = P_E \quad (9.17)$$

Thấy rõ bộ ước tính xác suất lỗi Monte Carlo là loại *không chệch*.

Phương sai của bộ ước tính Monte Carlo cho xác suất lỗi là:

$$\sigma_{\hat{P}_E}^2 = \frac{\sigma_{N_e}^2}{N^2} \quad (9.18)$$

Thay (9.15) vào (9.18) ta có:

$$\sigma_{\hat{P}_E}^2 = \frac{P_E(1-P_E)}{N} \quad (9.19)$$

Cho thấy bộ ước tính là loại *kiên định* khi $N \to \infty$. Cần lưu ý rằng, cả (9.17) và (9.19) đều nhận phân bố nhị thức cơ bản và chỉ đúng nếu các sự kiện lỗi là độc lập.

Khi dùng mô phỏng Monte Carlo để ước tính tham số của hệ thống truyền thông số chẳng như xác suất lỗi ký hiệu, thì ta mong muốn có được các ước tính không chệch và kiên định. Trên phương diện trung bình, nếu một bộ ước tính là không chệch, thì mô phỏng Monte Carlo cho ta kết quả chính xác. Ngoài ra, nếu một bộ ước tính là hữu hiệu, thì nó phải có phương sai nhỏ với xác suất lớn, thì ước tính đó nằm trong lân cận giá trị đúng. Nếu bộ ước tính không bị chệch và kiên định, thì cần phải môi phỏng nhiều lần truyền dẫn ký hiệu hơn để đếm được nhiều lỗi hơn trong một lần mô phỏng, làm giảm phương sai bộ ước tính. Phương trình (9.19) cho ta cảm nhận về số lỗi phải được đếm để ước tính có phương sai cho trước và thời gian mô phỏng cần thiết. Tuy nhiên, vấn đề thực tế đối với (9.19) là không thể sử dụng nó để xác định giá trị được yêu cầu của N đối với một phương sai cho trước vì trước khi mô phỏng ta chưa biết P_E. Tuy nhiên nhiều bài toán thực tế, ta có thể xác định giá P_E trong phạm vi thao tác hoặc ứng dụng các giới hạn hoặc các công cụ phân tích khác sao cho vẫn sử dụng được (9.19). Các bộ ước tính bị chệch nhưng lại có tính kiên định hội tụ vào giá trị sai, thì đó là điều không mong muốn trừ khi ta biết làm thế nào để loại bỏ sự chệch đó.

Mặc dù cần phải biết các đặc tính của bộ ước tính, nhưng trong nhiều trường hợp chỉ thấy, cho trước một bộ ước tính không chệch và kiên định là nhiệm vụ rất khó. Cần nhấn mạnh rằng tất cả các kết quả đạt được trong phần này, chỉ đúng nếu lỗi gây ra bởi tạp âm kênh là *độc lập* sao cho sự phân bố lỗi cơ bản là nhị thức. Nếu các sự kiện lỗi bị tương quan nhau (như trường hợp kênh bị hạn chế băng), thì các kết quả được đưa ra ở đây không còn hợp lệ nữa và ta phải giải quyết bài toán khó hơn nhiều. Tuy nhiên, nếu các sự kiện lỗi không độc lập nhưng (9.11) vẫn là bộ ước tính xác suất lỗi hợp lệ.

Ví dụ 9.2: Khi các sự kiện lỗi là *độc lập*, thì truyền dẫn nhị phân có thể được mô hình hóa như một thí nghiệm tung đồng xu. Khi này, truyền dẫn N ký tự được mô hình hóa bởi N lần tung đồng xu. Giả sử rằng kết cục xuất hiện "*mặt sấp*" tại lần tung thứ i tương ứng với một "*quyết định đúng*" trong lần truyền thứ i, và kết cục xuất hiện "*mặt ngửa*" ở lần tung thứ i tương ứng với "*lỗi*" xảy ra ở lần truyền thứ i. Trong ví dụ này, các thống kê liên quan với thí nghiệm tung đồng xu được xác định bằng mô phỏng. Vì việc tung đồng xu là độc lập, nên thí nghiệm này mô hình hóa cho truyền dẫn dữ liệu nhị phân trong kênh AWGN.

Giả sử rằng kết cục "*mặt sấp*" (không lỗi) xuất hiện với xác suất $(1-p)$ và kết cục "*mặt ngửa*" (có lỗi) xuất hiện với xác suất p, và ta muốn ước tính giá trị của p bằng cách tung đồng xu N lần. Ước tính Monte Carlo của p là:

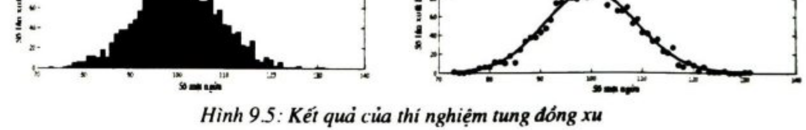
$$\hat{p} = \frac{N_{\text{ngửa}}}{N} \Leftrightarrow \frac{N_{\text{lỗi}}}{N} \tag{9.20}$$

Trong đó $N_{\text{ngửa}}$ biểu thị cho số "*mặt ngửa*" xuất hiện trong M lần tung. Tất nhiên, với N lần tung, giá trị của $N_{\text{ngửa}}$ có thể thay đổi từ $0 \rightarrow N$ nhưng xác suất xuất hiện k lần "*mặt ngửa*" trong N lần tung là:

$$\underbrace{p_N(k)}_{\substack{\text{xác suất xảy ra} \\ \text{k lần mặt ngửa} \\ \text{trong N lần tung}}} = \underbrace{\binom{N}{k}.p^k.(1-p)^{N-k}}_{p \text{ là xác suất xuất hiện mặt ngửa (lỗi)}} \tag{9.21}$$

Vì vậy ta phải thực hiện thí nghiệm này N lần để ước tính phân bố thống kê của $N_{\text{ngửa}}$, và xác định ước tính $\hat{p}$ của p. Chương trình Matlab mô phỏng thí nghiệm tung đồng xu được cho bởi **NVD9_cointoss.m** (có trong Phụ lục 9A)

Kết quả chạy chương trình Matlab **NVD9_cointoss.m** được cho ở hình 9.5. Cho thấy đồ thị (hình 9.5(a)) và các kết cục của các thí nghiệm riêng biệt, cùng với kết quả lý thuyết được minh họa (hình 9.5(b)). Lưu ý rằng kết quả lý thuyết là xấp xỉ Gausơ. Hệ số nhị thức được tính bằng cách sử dụng hàm **NVD9_nkchoose.m**.



Hình 9.5: Kết quả của thí nghiệm tung đồng xu

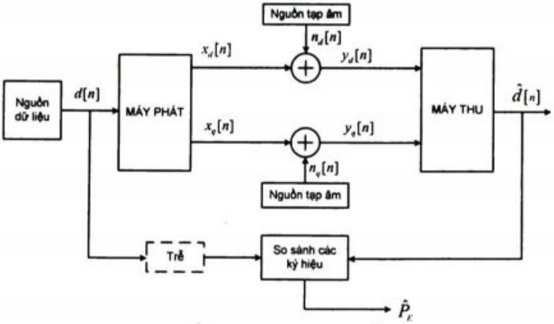
Hệ số nhị thức được tính theo cách này nhằm minh họa thuật toán được xem là hữu hiệu khi các giá trị của *k* lớn. Mặc dù Matlab có dải động lớn, và kỹ thuật đã được minh họa ở các mã chương trình trên không cần thiết cho ví dụ này, nhưng hữu hiệu khi dùng các ngôn ngữ khác.

9.3.2. Hai mô phỏng Monte Carlo đơn giản

Phần này, ta dùng phương pháp Monte Carlo để mô phỏng hệ thống truyền thông với những giả định sau:

- *Không thực hiện định dạng xung ở máy phát.*
- *Kênh là kênh AWGN*
- *Các ký hiệu dữ liệu tại đầu ra của nguồn là độc lập nhau và đồng xác suất.*
- *Hệ thống không lọc và không có giao thoa giữa các ký hiệu ISI.*

Các giả định này làm đơn giản hệ thống và chương trình mô phỏng. Dễ dàng xử lý các hệ thống theo cách giải tích (hệ thống thuộc loại xử lý được theo phép giải tích, liên hệ với chương 1) và dễ dàng có được xác suất lỗi.



Hình 9.6: Mô hình mô phỏng hệ thống truyền thông đơn giản

Mặc dù các ví dụ dưới đây đơn giản nhưng quan trọng khi quan trắc và hữu hiệu cho phần sau. Ngoài ra, nó thiết lập cấu trúc cơ bản của một chương trình mô phỏng cơ bản. Ta sẽ thấy tính cách của các mô phỏng Monte Carlo khi áp dụng vào các bài toán được chú trọng vào các chủ đề nghiên cứu, cụ thể là những hệ thống viễn thông số. Mô hình mô phỏng cơ bản được minh họa ở hình 9.6. Do không lọc nên trễ qua hệ thống bằng 0. Vì vậy, khối trễ (được đề cập ở chương 1) được nối tắt. Tuy nhiên, khối trễ vẫn được thấy ở hình 9.6 nhằm nhấn mạnh rằng cần phải có phần tử quan trọng trong hầu hết các mô phỏng.

Do các giả định trên, nên chỉ có duy nhất một nguồn gây ra lỗi là kênh tạp âm. Vì vậy, ta chọn giải pháp xác định các thành phần đồng pha và vuông pha xd(t) và xq(t) để chỉ rõ các thành phần không gian tín hiệu của tín hiệu chứ không phải là các mẫu của dạng sóng trong miền thời gian. Ưu điểm của giải pháp này là các thành phần không gian tín hiệu có thể được chỉ rõ bằng cách chỉ cần dùng ***một*** mẫu trên ký hiệu phát. Việc xử lý mô phỏng theo từng mẫu/ký hiệu mang lại thời gian mô phỏng nhanh.

Theo giải pháp này, tín hiệu thông dải tại đầu ra bộ điều chế được biểu diễn ở dạng:

$$x(t,n) = A_c \cos\left[2\pi f_c t + k_m d[n] + \theta\right] \quad (9.22)$$

Trong đó A_c là biên độ sóng mang, k_m là hằng số phụ thuộc điều chế, *d[n]* là ký hiệu dữ liệu thứ *n* (*d[n]* = 0 hoặc 1), θ là pha tham chiếu. Theo đó, đường bao phức $\tilde{x}[n]$ của *x(t,n)* chỉ còn là một hàm của chỉ số *n* và được cho bởi:

$$\tilde{x}[n] = A_c\, e^{k_m d[n] + \theta} \quad (9.23)$$

Các ví dụ được xét ở đây ta đều giả thiết $\theta = 0$. Vì vậy, trong hình 9.6 có các thành phần đồng pha và vuông pha:

$$x_d[n] = A_c.\cos\left(k_m d[n]\right) \quad (9.24)$$

$$x_q[n] = A_c \sin\left(k_m d[n]\right) \quad (9.25)$$

Để xác định và vẽ đồ thị BER là một hàm của E_b/N_0 cho hệ thống được minh họa ở hình 9.6, giá trị của E_b được giữ không đổi và công suất nhiễu được gia tăng trong dải xét. Muốn vậy, phải điều chỉnh công suất tạp âm tại đầu ra bộ tạo tạp âm trong hình 9.6. Từ chương 7 cho thấy, quan hệ giữa phương sai tạp âm và mật độ phổ công suất tạp âm (PSD) được cho bởi:

$$\sigma_n^2 = \frac{N_0 f_s}{2} \quad (9.26)$$

$$\Rightarrow N_0 = \sigma_n^2 \times \frac{2}{f_s} \quad (9.27)$$

Tỷ số tín hiệu trên tạp âm *SNR* được định nghĩa là E_b/N_0 trong đó f_s là tần số lấy mẫu. Vì vậy:

$$SNR = \frac{f_s}{2} \times \frac{E_b}{\sigma_n^2} \quad (9.28)$$

Nếu chuẩn hóa năng lượng E_b và tần số lấy mẫu f_s, thì ta có:

$$\sigma_n = \sqrt{\frac{1}{2} \times \frac{1}{SNR}} \quad (9.29)$$

Biểu thức này được sử dụng để thiết lập độ lệch chuẩn tạp âm trong mô phỏng dưới đây.

Ví dụ 9.3: Khóa dịch pha nhị phân, BPSK. Để tạo các thành phần không gian tín hiệu đồng pha và vuông pha của tín hiệu BPSK, ta cho $A_c = 1$ và $k_m = \pi$ ở (9.24) và (9.25). Ta được:

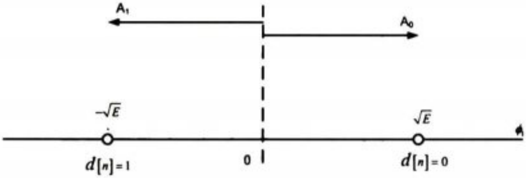
$$x_d[n] = \cos\left(\pi.d[n]\right) = \begin{cases} 1, & d[n] = 0 \\ -1, & d[n] = 1 \end{cases} \quad (9.30)$$

$$x_q[n] = \sin\left(\pi.d[n]\right) = \begin{cases} 0, & d[n] = 0 \\ 0, & d[n] = 1 \end{cases} \quad (9.31)$$

$x_q(t) = 0$ trong cả hai trường hợp *d[n]* = 0 và *d[n]* = 1. Theo đó, biểu diễn không gian tín hiệu BPSK được cho ở hình 9.7, trong đó ϕ_1 là hàm cơ sở của không gian tín hiệu. Vì không gian tín hiệu là một chiều (được tạo ra bởi một hàm cơ sở) nên chỉ cần tạo thành phần đồng pha

của tín hiệu và tạp âm trong mô phỏng, khi này loại bỏ thành phần vuông pha. Hình 9.7 cũng minh họa vùng quyết định A_0 và A_1. Nếu điểm tín hiệu thu rơi vào trong vùng A_0 (vùng bên phải) thì máy thu quyết định $\hat{d}[n] = 0$. Nếu điểm tín hiệu thu rơi vào trong vùng A_1 (vùng bên trái) thì máy thu quyết định $\hat{d}[n] = 1$. Ngưỡng thu là 0 vì các tín hiệu đồng xác suất, năng lượng như nhau, môi trường kênh AWGN. Vì vậy, nguyên tắc quyết định là:

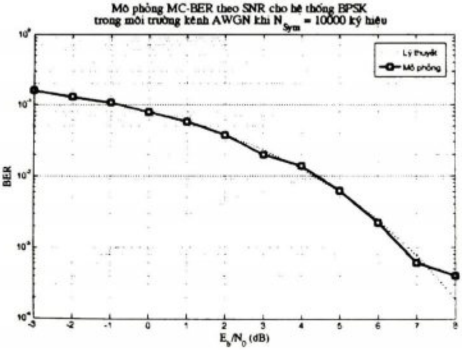
$$\hat{d}[n] = \begin{cases} 0, & y_d[n] > 0 \\ 1, & y_d[n] < 0 \end{cases} \tag{9.32}$$



Hình 9.7: Biểu diễn không gian tín hiệu của tín hiệu BPSK

Chương trình mô phỏng Matlab thực hiện bài toán này được cho bởi **NVD9_MCBPSK.m** (có trong Phụ lục 9A)

Chạy chương trình Matlab, N = 1000 ký hiệu cho mỗi giá trị của SNR, kết quả được minh họa ở hình 9.8. Lưu ý rằng, độ tin cậy của mô phỏng BER giảm khi SNR tăng lên do thực tế đếm được ít lỗi hơn. Điều này gợi ý rằng có thể có quan hệ giữa số ký hiệu được mô phỏng với SNR hoặc tiếp tục thực hiện mô phỏng đến khi số lỗi đếm được là giống nhau tại mỗi giá trị của SNR.

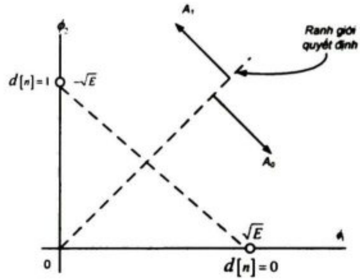


Hình 9.8: Kết quả mô phỏng BER hệ thống BPSK

Ví dụ 9.4: Khóa dịch tần nhị phân BFSK. Để tạo thành phần đồng pha và vuông pha của không gian tín hiệu BFSK, ta đặt $k_m = \pi/2$ trong (9.24) và (9.25), khi này:

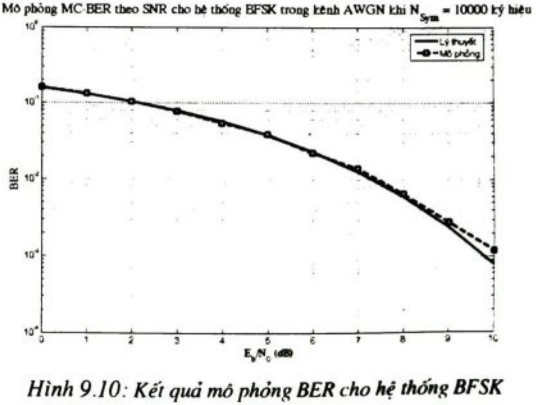
$$x_d[n] = \cos\left(\frac{\pi}{2}.d[n]\right) = \begin{cases} 1, & d[n]=0 \\ 0, & d[n]=1 \end{cases} \quad (9.33)$$

$$x_q[n] = \sin\left(\frac{\pi}{2}.d[n]\right) = \begin{cases} 0, & d[n]=0 \\ 1, & d[n]=1 \end{cases} \quad (9.34)$$



Hình 9.9: Biểu diễn không gian tín hiệu cho tín hiệu BFSK

Suy ra không gian tín hiệu của BFSK được cho ở hình 9.9, trong đó ϕ_1 và ϕ_2 là các hàm cơ sở của không gian tín hiệu (nhớ lại từ chương 4 đối với không gian hai chiều, các hàm cơ sở được thấy khi định nghĩa các thành phần đồng pha và vuông pha của tín hiệu đường bao phức thông thấp). Vì không gian tín hiệu BFSK là hai chiều, nên trong mô phỏng phải tạo thành phần đồng pha và vuông pha của tín hiệu BFSK và tạp âm. Hình 9.9 minh họa các vùng quyết định. Nếu điểm tín hiệu thu rơi vào trong vùng A_0, thì máy thu quyết định $\hat{d}$ [n] = 0. Nếu điểm tín hiệu thu rơi vào trong vùng A_1, thì máy thu quyết định $\hat{d}$ [n] = 1. Lưu ý rằng, khi cho trước một điểm trong không gian tín hiệu thể hiện một tín hiệu thu ($y_d[n]$ và $y_q[n]$ trong hình 9.9), thì luật quyết định là:



Hình 9.10: Kết quả mô phỏng BER cho hệ thống BFSK

$$\hat{d}[n]=\begin{cases}0, & y_d[n]>y_q[n]\\ 1, & y_d[n]<y_q[n]\end{cases} \tag{9.35}$$

Chương trình Matlab thực hiện mô phỏng được cho bởi **NVD9_MCBFSK.m** (có trong phụ lục 9A). Chạy chương trình Matlab với thông số N = 10.000 ký hiệu cho mỗi giá trị của SNR, kết quả được cho ở hình 9.10. Một lần nữa lưu ý rằng, độ tin cậy của ước tính *giảm* khi SNR *tăng* (do thực tế đếm được ít lỗi hơn). Cần phải có các hoạt động hiệu chỉnh thích hợp như ví dụ trước.

9.4. Tích phân Monte Carlo

Tích phân Monte Carlo là chủ đề rất quan trọng trong nghiên cứu viễn thông. Nhớ lại rằng, trong môi trường kênh AWGN, con số V được tạo ra bằng cách lấy mẫu tín hiệu đầu ra của bộ tách sóng tích hợp và kết xuất (*integrate-and-dump detector*) là một biến ngẫu nhiên Gausơ có trung bình được xác định là giá trị ký hiệu dữ liệu và phương sai được xác định bởi tạp âm kênh. Các hàm mật độ xác suất có điều kiện pdf, được điều kiện hóa trên $d[n] = 0$ và $d[n] = 1$ được minh họa ở hình 9.11, trong đó k_T là ngưỡng thu. Xác suất lỗi có điều kiện với điều kiện $d[n] = 1$ là:

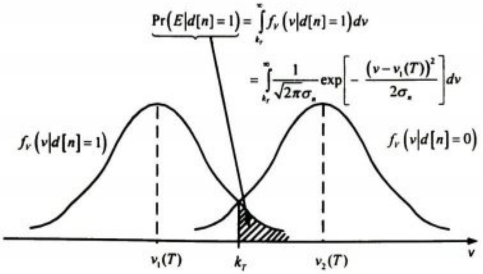
$$\Pr\Big(\underset{\text{lỗi}}{E}\big|\underbrace{d[n]=1}_{\text{phát bit 1}}\Big)=\int_{k_T}^{\infty}\frac{1}{\sqrt{2\pi}\sigma_n}e^{-\frac{(v-v_1(T))^2}{2\sigma_n}}dv \tag{9.36}$$

Tương tự ta có biểu thức $\Pr\big(E|d[n]=0\big)$. Theo đó, ước tính xác suất lỗi của hệ thống là:

$$P_E=\frac{1}{2}\Pr\Big(E|\underbrace{d[n]=1}_{\text{phát bit 1}}\Big)+\frac{1}{2}\Pr\Big(E|\underbrace{d[n]=0}_{\text{phát bit 0}}\Big) \tag{9.37}$$

Bao gồm ước tính giá trị của nguyên hàm.

Nghiên cứu tích phân Monte Carlo giúp hiểu sâu hơn về kỹ thuật mô phỏng Monte Carlo. Chẳng hạn, nghiên cứu tích phân Monte Carlo tạo ra tình huống đơn giản trong đó minh họa các tính chất hội tụ của một bộ ước tính Monte Carlo.



Hình 9.11: pdf có điều kiện đối với tín hiệu nhị phân trong kênh tạp âm Gausơ

9.4.1. Khái niệm cơ bản

Giả sử cần ước lượng tích phân:

$$I = \int_0^1 g(x)dx \tag{9.38}$$

Trong đó *g(x)* là một hàm được giới hạn trong phạm vi lấy tích phân. Từ lý thuyết xác suất cơ bản, giá trị kỳ vọng (trung bình toàn bộ) của hàm *g(x)* là:

$$E\{g(X)\} = \int_{-\infty}^{\infty} g(x)f_X(x)dx \tag{9.39}$$

Trong đó $f_X(x)$ là hàm mật độ xác suất của biến ngẫu nhiên *X*. Nếu pdf của *X* thỏa mãn $f_X(x) = 1$ trong khoảng (0,1) và bằng 0 ở các trường hợp còn lại, thì $E\{g(X)\} = I$. Vì vậy, nếu *U* là một biến ngẫu nhiên được phân bố đều trong khoảng (0,1) thì:

$$I = E\{g(U)\} \tag{9.40}$$

Sử dụng các đối số tần suất tương đối ta có thể viết:

$$\lim_{N\to\infty}\left[\frac{1}{N}\sum_{i=0}^{N} g(U_i)\right] = E\{g(U)\} = I \tag{9.41}$$

Vì vậy, ta mô phỏng hàm bị tích để lấy mẫu nó tại *N* điểm trong khoảng (0,1). Sau đó, giá trị trung bình của các mẫu cung cấp cho bộ ước tính để có được giá trị của tích phân. Mô phỏng Monte Carlo hệ thống làm giống như vậy. Vì thường không có biểu thức dạng kín cho con số trên vùng lỗi, nên các mẫu của con số đó được tạo ra bằng cách dùng mô phỏng hệ thống.

Nếu tính giới hạn (9.41) sai (nó sẽ luôn là một trường hợp trong những ứng dụng thực tế), chỉ nhận được các kết quả xấp xỉ. Ký hiệu phép xấp xỉ này là $\hat{I}$ ta có:

$$\frac{1}{N}\sum_{i=0}^{N} g(U_i) = \hat{I} \tag{9.42}$$

Cho bộ ước tính tích phân Monte Carlo. Tóm lại, bộ ước tính tích phân được thực hiện bằng cách ước lượng hàm *g(x)* tại *N* điểm ngẫu nhiên phân bố đều và lấy trung bình. Có thể áp dụng quá trình đó cho bất kỳ tích phân phù hợp. Bằng cách đổi biến số, các tích phân thích hợp có các giới hạn tùy ý được ước lượng bằng cách dùng các kỹ thuật Monte Carlo. Ví dụ: tích phân:

$$I = \int_a^b f(x)d(x) \tag{9.43}$$

Có thể chuyển về dạng chuẩn bằng cách đổi biến $y = {}^{(x-a)}/_{(b-a)}$:

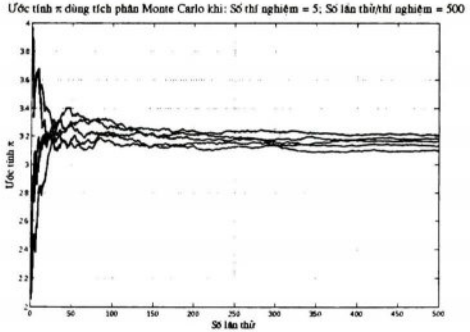
$$I = (b-a)\int_0^1 f\left[a+(b-a)y\right]dy \tag{9.44}$$

Ví dụ 9.5: **Để ước tính giá trị của π bằng tích phân Monte Carlo, chỉ cần tìm tích phân xác định mà giá trị của nó là một hàm đã biết của π.** Tích phân nhanh chóng trở thành:

$$I = \int_0^1 \frac{dx}{1+x^2} = \frac{\pi}{4} \tag{9.45}$$

Vì vậy, ta dùng thuật toán (9.42) để ước lượng tích phân I và nhân kết quả đó với 4. Hiển nhiên, tốt nhất ta sử dụng giá trị N lớn nhưng hữu hạn. Khi này, không đạt được π nhưng là một giá trị xấp xỉ với π. Vì vậy, giá trị của tích phân, giá trị ước tính của π là biến ngẫu nhiên (giá trị ngẫu nhiên). Các kết quả được cho ở hình 9.12 đối với 5 ước tính của π, mỗi ước tính dựa trên 500 cuộc *thử nghiệm*. Năm ước tính đó là:

$$\hat{\pi} = [3,1418\ \ 3,1529\ \ 3,1517\ \ 3,1040\ \ 3,1220] \tag{9.46}$$



Hình 9.12: Ước tính π dùng tích phân Monte Carlo

Nếu lấy trung bình 5 kết quả trên ta được:

$$\hat{\pi} = 3,1345 \tag{4.47}$$

Chương trình Matlab để ước tính π bằng tích phân Monte Carlo được cho bởi **NVD9_example5.m** (có trong Phụ lục 9A).

9.4.2. Tính hội tụ

Giả sử giá trị của tích phân I sẽ được ước tính và sẵn có N quan trắc ngẫu nhiên (hay các mẫu) được ký hiệu là X_i. Ta tạo bộ ước tính của I là:

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N} X_i \tag{9.48}$$

Giả sử N quan sát X_i là phân bố đồng nhất độc lập thống kê nhau. Trung bình số học của các mẫu được cho bởi:

$$E\left\{\frac{1}{N}\sum_{i=1}^{N} X_i\right\} = \frac{1}{N}\sum_{i=1}^{N} E\{X_i\} = \frac{NI}{N} = I \tag{9.49}$$

Sao cho ước tính $\hat{I}$ là ước tính không chệch. Vì các quan sát được giả định là độc lập, nên phương sai mẫu là:

$$\sigma_{\hat{I}}^2 = \frac{1}{N^2}\sum_{i=1}^{N}\sigma_x^2 = \frac{N\sigma_x^2}{N^2} = \frac{\sigma_x^2}{N} \tag{9.50}$$

Cho thấy bộ ước tính tích phân này là loại ước tính *kiên định*.

Giả sử $X_i = g(U_i)$, thì phương sai của các mẫu được ký hiệu là σ_x^2 và được cho bởi:

$$\sigma_x^2 = \int_0^1 g^2(u)du - \left[\int_0^1 g(u)du\right]^2 \tag{9.51}$$

Vì vậy, nếu cho trước hàm bị tích *g(u)*, thì có thể xác định được giá trị *N* đối với một phương sai lỗi cho trước. Vì bộ ước tính thuộc loại kiên định, nên ta sẽ đạt được các ước tính chính xác của tích phân *I* nếu *N* đủ lớn. Cũng theo đó, ta sẽ đạt được các ước tính chính xác của tích phân *I* nếu các mẫu của hàm $g(u_i)$ có phương sai nhỏ. Vì vậy, các ước tính Monte Carlo của một tích phân sẽ rất chính xác đối với một giá trị cho trước của N nếu $g(u_i)$ xấp xỉ hằng số (*mịn*) trên khoảng lấy tích phân. Mấu chốt của thực tế là, nếu $g(u_i)$ không đổi trên dải lấy tích phân, thì ước tính $\hat{I}$ của *I* là chính xác khi $N = 1$.

9.4.3. Khoảng tin cậy

Chất lượng của ước tính $\hat{I}$ thường được biểu diễn dưới dạng khoảng tin cậy, cho ta *xác suất* (1-α) của sự kiện mà các ước tính rơi vào *dải* $\left(\pm\beta\sigma_{\hat{I}}\right)$ cho trước của các giá trị. Vì vậy, *hai tham số cần lưu ý*: xác suất được cố định bởi α và dải được cố định bởi β. Ở dạng phương trình, khoảng tin cậy được định nghĩa bởi biểu thức:

$$\Pr\left\{I - \beta\sigma_{\hat{I}} \le \underbrace{\hat{I}}_{\text{ước tính của I}} \le I + \beta\sigma_{\hat{I}}\right\} = 1-\alpha \tag{9.52}$$

Như được mô tả ở hình 9.13. Ta coi khoảng $I \pm \beta\sigma_{\hat{I}}$ là khoảng tin cậy 1-α. Tại đây, ta xét giá trị của tham số β. Phương trình (9.52) được viết theo *lỗi* ($\hat{I} - I$) như sau:

$$\Pr\left\{-\beta\sigma_{\hat{I}} \le \underbrace{\hat{I} - I}_{\text{lỗi}} \le \beta\sigma_{\hat{I}}\right\} = 1-\alpha \tag{9.53}$$

Trong đó $\sigma_{\hat{I}} = \sigma_x/\sqrt{N}$ với σ_x được xác định theo (9.51). Giả sử lỗi ($\hat{I} - I$) là biến ngẫu nhiên Gausơ (giả định hợp lý khi *N* lớn), thì hàm mật độ xác suất của ($\hat{I} - I$) được tính xấp xỉ bởi:

$$\frac{1}{\sqrt{2\pi}\sigma_{\hat{I}}} e^{-\frac{(\hat{I}-I)^2}{2\sigma_{\hat{I}}^2}}$$

Lưu ý rằng, ($\hat{I} - I$) là biến ngẫu nhiên trung bình không do ước tính của tích phân không bị lệch. Theo đó:

$$\Pr\left\{\hat{I} - I \underbrace{\geq}_{note} \beta\sigma_{\hat{I}}\right\} = \frac{1}{\sqrt{2\pi}\sigma_{\hat{I}}} \int_{\beta\sigma_{\hat{I}}}^{\infty} e^{-\frac{t^2}{2\sigma_{\hat{I}}^2}} dt \tag{9.54}$$

Cùng với đổi biến số y = t/$\sigma_{\hat{I}}$ ta có:

$$\Pr\left\{\hat{I} - I \geq \beta\sigma_{\hat{I}}\right\} = \frac{1}{\sqrt{2\pi}} \int_{\beta}^{\infty} e^{\frac{y^2}{2}} dy = Q(\beta) \tag{9.55}$$

Trong đó Q(.) biểu diễn hàm Q Gausơ.

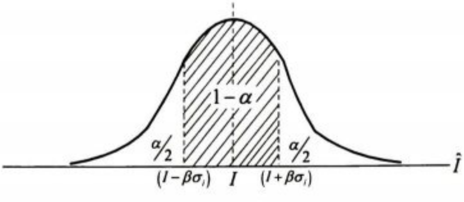
Từ hình 9.13 ta có :

$$\Pr\left\{\hat{I} - I \geq \beta\sigma_{\hat{I}}\right\} = Q(\beta) = \frac{\alpha}{2} \tag{9.56}$$

Vì vậy tìm được β là:

$$\beta = Q^{-1}\left(\frac{\alpha}{2}\right) \tag{9.57}$$

Vì vậy, như được cho ở hình 9.13, xác suất mà ước tính của I nằm trong khoảng $I \pm Q^{-1}\left(\alpha/2\right).\frac{\sigma_x}{\sqrt{N}}$ là $(1-\alpha)$, trong đó σ_x được cho bởi (9.51). Các đại lượng $\pm Q^{-1}\left(\alpha/2\right).\frac{\sigma_x}{\sqrt{N}}$ xác định giới hạn trên và giới hạn dưới. Để ước lượng được các đại lượng này, phải xác định được σ_x.



Hình 9.13: Khoảng tin cậy

Ví dụ 9.6: Để minh họa các khái niệm trên, ta xét tích phân:

$$I = \int_0^1 e^{-t^2} dx \tag{9.58}$$

Sử dụng tích phân số (chẳng hạn hàm **quad** trong Matlab) giá trị của *I* là:

$$I \approx 0{,}7468 \tag{9.59}$$

Tương tự:

$$I_2 = \int_0^1 \left(e^{-t^2}\right)^2 dt \approx 0,5981 \quad (9.60)$$

Thay (9.59) và (9.60) vào (9.51) ta có:

$$\sigma_x^2 = 0,5981 - (0,7468)^2 = 0,0404 \quad (9.61)$$

Vì vậy, độ lệch chuẩn ước tính của tích phân là:

$$\sigma_{\hat{I}} = \frac{\sigma_x}{\sqrt{N}} = \frac{0,2010}{\sqrt{N}} \quad (9.62)$$

và các giới hạn tin cậy trên và dưới là:

$$I \pm \beta\sigma_{\hat{I}} = 0,7468 \pm Q\left(\frac{\alpha}{2}\right)^{-1} \cdot \left(\frac{0,2010}{\sqrt{N}}\right) \quad (9.63)$$

Kết quả được minh họa trong hình 9.14. Chương trình Matlab để tạo ra những kết quả hình 9.14 được cho ở **NVD9_examle6.m** (có trong Phụ lục 9A).

Chương 10

MÔ PHỎNG MONTE CARLO HỆ THỐNG TRUYỀN THÔNG

10.1. Mở đầu

Chương này ta mở rộng các kỹ thuật Monte Carlo (MC) cơ bản đã được đề cập ở chương 9. Ta sẽ mô phỏng hai hệ thống viễn thông điển hình: *(i)* Hệ thống truyền thông số khóa dịch pha (PSK), mặc dù rất đơn giản nhưng nó tạo ra các khối cơ bản để phát triển mô phỏng cho các hệ thống phức tạp; *(ii)* Hệ thống QPSK vi sai, trong đó ta sẽ xét đến các ảnh hưởng của lỗi đồng bộ ký hiệu và lỗi pha. Chương này ta cũng sẽ đề cập kỹ thuật bán phân tích (SA), là kỹ thuật kết hợp mô phỏng Monte Carlo và phân tích.

Hai phương pháp luận được đề cập trong chương này hơi khác. *(i)* Mô phỏng Monte Carlo yêu cầu phân tích toán học rất ít và có thể được áp dụng cho hệ thống truyền thông bất kỳ, trong đó cần có các thuật toán xử lý tín hiệu để biểu diễn mỗi khối chức năng. Theo đó, mô phỏng Monte Carlo là *công cụ tổng quát* nhưng thời gian mô phỏng rất lâu, vì vậy như đã thấy ở chương trước, cần phải dung hòa *giữa* tính chính xác và thời gian mô phỏng; *(ii)* Mô phỏng bán phân tích SA yêu cầu mức độ phân tích nhiều hơn, nhưng bù lại giảm được thời mô phỏng. Ngoài ra, khi thực hiện *một* mô phỏng MC đem lại ước tính BER tại *một* giá trị của E_b/N_0, trong khi đó mô phỏng bán phân tích SA cho ta *một đường cong đầy đủ* của BER, nó là một hàm của E_b/N_0. Tuy nhiên, ta sẽ thấy rằng mô phỏng bán phân tích SA không phải là phương pháp luận mà có thể được áp dụng phổ biến.

Để được tường minh quá trình thực thi hóa mô hình mô phỏng bằng chương trình Matlab như: quá trình tạo và biến đổi tín hiệu, tham số vào/ra của các khối chức năng cũng như việc phù hợp hóa các tham số tại các điểm liên kết các khối chức năng, các tham số đặc trưng của từng khối chức năng,... cần phải nghiên cứu kĩ mã chương trình Matlab đặc biệt tính modul của mã chương trình mô phỏng. Trên cơ sở đó có thể: *(i)* Khảo sát chi tiết các tín hiệu tại các điểm khác nhau của hệ thống trong các miền thời gian và tần số bằng các bộ FFT và IFFT kết hợp cùng với các bộ hậu xử lý như các hàm vẽ, các biểu đồ mắt, chòm sao tín hiệu, các biểu đồ tán xạ... muốn vậy cần phải biết cách lấy các dự liệu của các tín hiệu tại các điểm khác nhau của chương trình mô phỏng (tương ứng với mô hình mô phỏng). *(ii)* Nghiên cứu khảo sát, quan sát các tín hiệu trong các miền thời gian và tần số cũng như hiệu năng của hệ thống dưới các ảnh hưởng khác nhau tham số kênh cũng như các tham số đặc trưng của từng khối chức năng. ***(iii)*** Cập nhật, nâng cấp mô hình thành mô hình riêng.

Cần phải xác định các rõ trong các chương trình mô phỏng: đâu là các hàm có sẵn trong thư viện Matlab và cách thức dùng vào ứng dụng cụ thể, đâu là các hàm được viết dành riêng cho chương trình các hàm này được cho ở các phụ lục A của các chương được đánh số theo mã số của chương.

10.2. Hai mô phỏng Monte Carlo điển hình

Như đã thấy ở chương trước, dùng kỹ thuật MC để ước tính BER bằng cách truyền N ký hiệu dữ liệu qua mô hình mô phỏng hệ thống và đếm số lỗi. Giả sử, ta truyền N ký hiệu qua mô hình mô phỏng đếm được N_e lỗi thì ước tính của BER là:

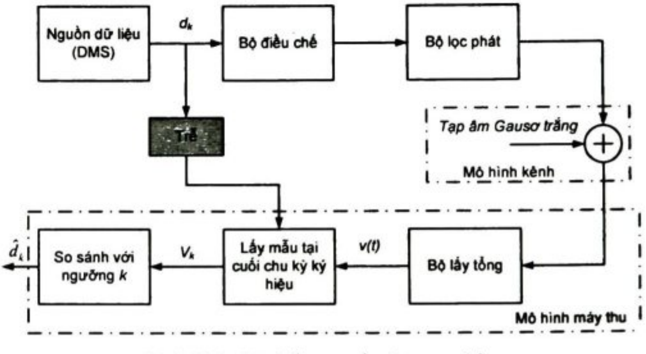
$$\hat{P}_E = \frac{N_e}{N} \tag{10.1}$$

Từ chương trước, $\hat{P}_E$ là một biến ngẫu nhiên và ước tính chính xác của BER yêu cầu bộ ước tính $\hat{P}_E$ là loại *không chệch* và có *phương sai nhỏ*. Để phương sai nhỏ thì N lớn dẫn đến thời gian chạy mô phỏng lâu. Dưới đây ta minh họa kỹ thuật Monte Carlo bởi hai mô phỏng điển hình.

10.2.1. Mô phỏng Monte carlo hệ thống truyền thông BPSK cơ bản

- ***Mô hình mô phỏng***: Mô hình mô phỏng hệ thống truyền thông BPSK cơ bản được minh họa ở hình 10.1.

- ***Các giả định***: ***(i)*** Điều chế BPSK, có hai điểm tín hiệu nằm trên kênh đồng pha (xem lại ví dụ 9.3). Vì vậy, ta loại bỏ kênh vuông pha ra khỏi mô phỏng; ***(ii)*** Bộ lọc tại đầu ra bộ điều chế là một bộ lọc Butterworth bậc 3 có độ rộng băng tần bằng tốc độ bit ($BW = r_b$) nhằm thể hiện việc gây ra giao thoa giữa các ký hiệu (ISI); ***(iii)*** Kênh là kênh AWGN.



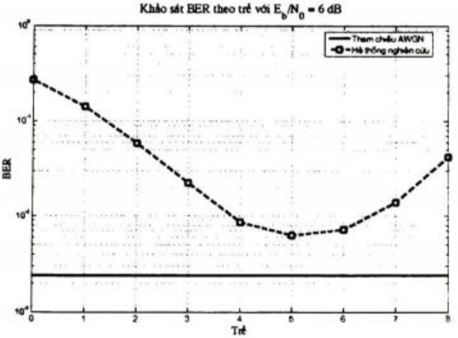
Hình 10.1: Hệ thống truyền thông cơ bản

- ***Mục đích nghiên cứu - chương trình và quá trình mô phỏng***: Mục đích của mô phỏng này là xác định sự tăng lên của BER do ISI gây ra từ bộ lọc. Chương trình mô phỏng hệ thống được cho ở file **NVD10_MCBPSKber.m** trong Phụ lục 10A. Sử dụng giải pháp *chuỗi-khối*, trong đó các khối dài 1000 ký hiệu được xử lý lặp cho đến khi toàn bộ N ký hiệu được xử lý. Nó được thực hiện đầu tiên để bộ lọc thường trình Matlab (là một hàm có sẵn trong thư viện Matlab) thực hiện tích chập miền thời gian. Khi dùng các hàm sẵn có trong thư viện Matlab, thì rất hiệu quả và giảm đáng kể thời gian chạy mô phỏng. Lưu ý rằng, phải đảm bảo đầu ra bộ lọc

là liên tục từ khối tới khối, nó được thực hiện bằng cách dùng tham số khởi đầu được đưa ra trong bộ lọc.

Trước hết, ta phải xác định giá trị của ***trễ***. Có nhiều cách, cách được dùng nhiều nhất là lấy tương quan chéo giữa đầu ra bộ điều chế và đầu ra máy thu, như đã được thực hiện ở chương 8 khi ta ước tính SNR. Phương pháp này sẽ được dùng khi xét mô phỏng bán phân tích SA. Để minh họa tầm quan trọng việc chọn chính xác giá trị của ***trễ***, trong trường hợp này ta sử dụng kỹ thuật khác. Cụ thể, ta chọn một giá trị E_b/N_0, ta mô phỏng hệ thống với các giá trị trễ khác nhau và quan sát kết quả. Chương trình Matlab tương ứng được cho ở file **NVD10_ MCBPSKdelay.m** (có trong Phụ lục 10A).

Lưu ý rằng, chương trình Matlab **NVD10_ MCBPSKdelay.m** là sự kết hợp giữa bộ tiền xử lý và bộ hậu xử lý. (cần phải nghiên cứu kỹ hàm **NVD10_MCBPSKrun.m** trong phụ lục 10A). Giá trị giả định của E_b/N_0 là 6dB và *trễ* được lặp từ 0 đến 8 mẫu. Vì tần số lấy mẫu là 10 mẫu/1 ký hiệu, nên kích cỡ bước của trễ là $0,1T_s$, với T_s là thời gian ký hiệu. Giá trị của N được lựa chọn sao cho số lỗi xảy ra phù hợp để đảm bảo phương sai bộ ước tính đủ nhỏ. Trong trường hợp này, ta đặt $N = 100/P_T$ với P_T là xác suất lỗi lý thuyết (tính toán) trong môi trường kênh AWGN. Tất nhiên, khi có ISI và những nhiễu loạn khác sẽ làm gia tăng lỗi đối với một giá trị của E_b/N_0. Lưu ý rằng, với mỗi giá trị của *trễ*, thì cả giá trị của BER và số lỗi để tính toán BER đều được hiển thị. Điều này cho phép kiểm tra giả định về "*giá trị thích hợp của lỗi để tạo ra trong một ước tính BER khả tin*".



*Hình 10.2: Mô phỏng khởi đầu để xác định **trễ** đối với hệ thống BPSK*

Kết quả mô phỏng được minh họa ở hình 10.2. Các kết quả mô phỏng được minh họa bằng hình quả trám, hiệu năng hệ thống lý tưởng (không có ISI, khi hệ thống hoạt động trong môi trường kênh AWGN với $E_b/N_0 = 6\ dB$) được cho bởi đường liền nét. Thấy rõ ràng, việc chọn giá trị của *trễ* không đúng sẽ làm giá trị của BER quá lớn. Vì giá trị của BER nhỏ nhất tại giá trị của trễ là 5 mẫu, nên có thể lấy 5 mẫu làm giá trị thích hợp của trễ. Tuy nhiên, vì trễ là số nguyên lần chu kỳ lấy mẫu, nên giá trị 5 có thể không là chính xác hoàn toàn. Quan sát hình 10.2 cho thấy rằng giá trị trễ chính xác là khoảng giữa 5 và 6 chu kỳ lấy mẫu. Việc ước tính giá

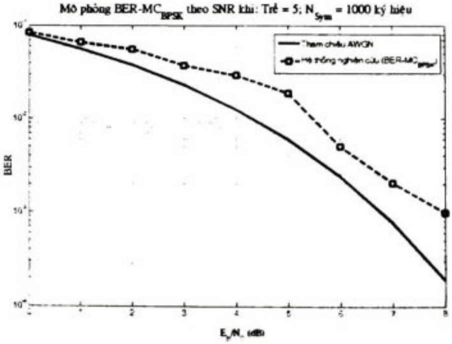
trị đúng của trễ được bằng mô phỏng *càng chính xác* thì tần số lấy mẫu *càng lớn hơn* (chu kỳ lấy mẫu nhỏ hơn). Cần lưu ý rằng, bộ ước tính được định nghĩa bởi công thức (10.1) là biến ngẫu nhiên, vì vậy bất kỳ giá trị cho trước của BER đều có thể quá lớn hoặc quá nhỏ. Bộ lọc phát gây ra ISI, và ảnh hưởng của ISI lên hiệu năng BER từ điểm giới hạn ISI = 0 đối với mỗi giá trị của trễ. Nếu bộ lọc phát được loại bỏ (bằng cách đặt tham số trong chương trình mô phỏng FilterSwitch = 0), thì ISI = 0.

Tiếp đó, sau khi biết giá trị của ***trễ***, ta thực hiện mô phỏng và xác định giá trị của $\hat{P}_E$, nó là một hàm của E_b/N_0 (nghĩa là khảo sát BER theo SNR). Chương trình Matlab tương ứng được cho ở file **NVD10_ MCBPSKber.m** (có trong Phụ lục 10A).

Lưu ý rằng, E_b/N_0 được tăng từng 1 dB, 0 dB đến 8 dB.

Khi thực hiện một mô phỏng Monte Carlo trên dải giá trị của E_b/N_0, thì với *cùng* một giá trị của N khi E_b/N_0 tăng sẽ đếm được ít lỗi hơn. Kết quả là, ước tính BER tại giá trị E_b/N_0 lớn sẽ kém tin cậy hơn so với ước tính BER tại giá trị E_b/N_0 nhỏ. Có thể giải quyết vấn đề này một cách từng phần bằng cách đạt $N = K/P_T$ (N là số mẫu được xử lý, P_T là xác suất lỗi cho trường hợp kênh AWGN). Do sự suy thoái của hệ thống (ISI, lỗi đồng bộ) dẫn đến các giá trị được mô phỏng của $\hat{P}_E > P_T$, có nhiều hơn K lỗi quan sát được trong chạy mô phỏng. Mã chương trình Matlab **NVD10_MCBPSKber.m** sử dụng K = 20. Vì mô phỏng dựa trên việc xử lý tuần tự các khối mẫu, với kích cỡ khối 1000 ký hiệu (10.000 mẫu), nên phải đảm bảo $N > 1000$ để ít nhất có một khối được xử lý. Nếu $N < 1000$ sẽ xảy ra kết quả không chính xác.

Kết quả chạy chương trình mô phỏng **NVD10_MCBPSKber.m** được minh họa ở hình 10.3. Các kết quả mô phỏng được biểu thị bằng các hình vuông, và BER lý tưởng (ISI = 0) được biểu thị bằng đường liền nét. BER tăng do ISI sinh ra bởi bộ lọc là hiển nhiên.

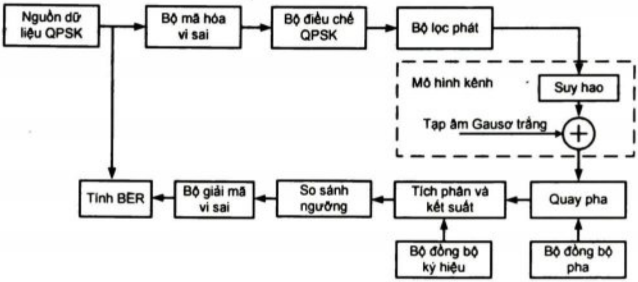


Hình 10.3: Mô phỏng BPSK đối với hệ thống có ISI

Kỹ thuật *chuỗi-khối* được dùng trong trường hợp này sẽ được sử dụng lại ở chương 18 khi đơn giản hóa mô phỏng hệ thống CDMA. Mô phỏng này, tuy đơn giản nhưng rất ý nghĩa ở chỗ, nó tạo ra các khối chức năng căn bản cho mô phỏng.

10.2.2. Mô phỏng Monte carlo hệ thống truyền thông QPSK cơ bản

Ở hệ thống đơn giản trên, ta đã mô phỏng hệ thống điều chế BPSK với nhiều giả định đơn giản. Trường hợp hệ thống QPSK, ta mô hình hóa một vài nguồn lỗi mới, một số tham số mới, với mục đích liên hệ các kết quả mô phỏng với các kết quả hệ thống truyền thông vật lý. Lưu ý rằng, trường hợp này suy hao kênh (suy hao truyền lan sóng giữa máy phát và máy thu) được đưa vào chương trình mô phỏng ở dạng tham số mô phỏng. Tốc độ ký hiệu và tần số lấy mẫu cũng được tính đến trong mô phỏng. Sơ đồ khối hệ thống được cho ở hình 10.4, và mã chương trình mô phỏng được cho ở file **NVD10_MCQPSKber.m** (trong Phụ lục 10A). Lưu ý rằng, bộ lọc phát, mặc dù có trong hình 10.4 nhưng không được dùng trong trường hợp mô phỏng này nhằm giảm thời gian mô phỏng. Bộ lọc phát được xét đến trong mã mô phỏng được cho ở phụ lục 10A.



Hình 10.4: Hệ thống truyền thông QPSK

Trong các hệ thống vô tuyến nhất quán, máy thu phải có khả năng đồng bộ ký hiệu và sóng mang. Tạp âm và méo kênh làm cho việc định thời và khôi phục sóng mang không hoàn hảo. Đồng bộ sóng mang không chính xác gây ra lỗi pha, làm xoay pha tín hiệu thu so với tín hiệu phát. Mô phỏng trong trường hợp này cho phép ta mô phỏng lỗi pha như là một quá trình stochastic. Lỗi đồng bộ ký hiệu làm cho bộ tách sóng tích phân và kết xuất (*integrate-and-dump*) xử lý tín hiệu thu trên một khoảng thời gian không chính xác. Vì vậy, chương trình mô phỏng cũng cho phép nghiên cứu lỗi từ ảnh hưởng này.

Từ lý thuyết truyền thông cơ bản cho thấy, các hệ thống QPSK bị ảnh hưởng từ sự không rõ ràng về pha. Do kênh tạo ra trễ ngẫu nhiên lên tín hiệu, nên máy thu không thể xác định được giá trị pha tuyệt đối của tín hiệu phát. Ví dụ, máy phát có thể gửi chuỗi pha 45^0, 135^0, 45^0 và -45^0. Giả sử kênh gây ra một lượng trễ là 100,75 chu kỳ sóng mang RF. Thì máy thu sẽ tách tín hiệu gốc 45^0 *thành* -45^0 và gây lỗi lên các ký hiệu còn lại, tạo ra chuỗi thu là -450, -135^0 và 135^0. Nếu các bit tin được truyền đi ở dạng pha tuyệt đối của tín hiệu phát, thì máy thu sẽ tạo ra số lượng lỗi lớn (nguồn gốc lỗi có thể do kênh). Giải pháp cho vấn đề này là mã hóa thông tin trước khi được truyền đi, thay bít tin được truyền đi ở dạng pha tuyệt đối của tín hiệu phát thì dùng sự sai khác pha giữa các ký hiệu. Ví dụ, nếu pha tín hiệu phát tăng 90^0 từ 45^0 đến 135^0 giữa ký hiệu thứ nhất và ký hiệu thứ hai, thì máy thu sẽ tách hai tín hiệu đó tại -135^0 và 45^0, mà

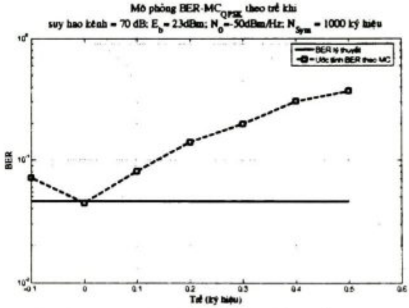
nó vẫn cho thấy tăng pha là 90^0. Mã hóa vi sai được thực hiện trong mã chương trình Matlab được cho trong phụ lục 10A, là mã mô phỏng chính cho hệ thống QPSK.

Cũng như hệ thống BPSK, trễ thời gian qua hệ thống phải được xác định. Chương trình Matlab **NVD10_MCQPSKdelay.m**, dưới đây ta xác định trễ thời gian tối ưu tại máy thu để giải thích cho trễ truyền tín hiệu qua hệ thống.

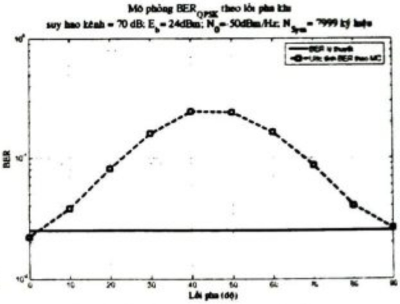
Vì không dùng bộ lọc kênh, nên trễ tối ưu là những ký hiệu 0, như được thấy trong hình 10.5. Lưu ý rằng, ta đã đo trễ theo chu kỳ ký hiệu chứ không phải là theo các mẫu như trường hợp trước.

Tại đây, biết được trễ, ta sẽ đánh giá mức độ nhạy cảm của BER đối với lỗi pha đồng bộ tĩnh. Lỗi pha được đo từ 0 đến 90^0 (tăng dần từng 10^0). Mã chương trình Matlab được cho ở **NVD10 _MCQPSKphasesync.m** (có trong Phụ lục 10A).

Kết quả chạy chương trình mô phỏng **NVD10_MCQPSKphasesync.m** cho phép khảo sát tính nhạy cảm của BER đối với các lỗi pha tĩnh, được minh họa ở hình 10.6, kết quả cho thấy BER có giá trị lớn nhất tại lỗi pha 45^0 và giảm dần tới giá trị tối ưu (không có lỗi pha) là 0 hoặc 90^0. Tính cách này có được do bộ mã hóa vi sai đem lại.



Hình 10.5: Mô phỏng khởi đầu để xác định trễ đối với hệ thống QPSK

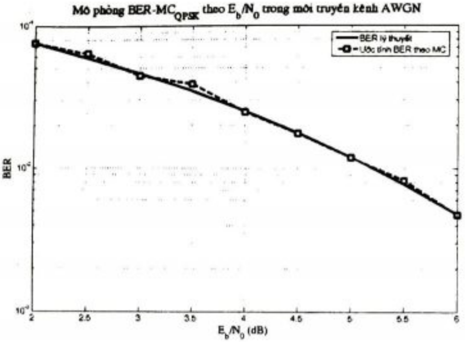


Hình 10.6: Minh họa tính nhạy của BER đối với lỗi pha tĩnh

Tại đây, ta biết được thời gian trễ và sự dịch pha tối ưu đối với kênh, ta tiến hành đo BER là một hàm của SNR (mô phỏng hiệu năng BER theo SNR). Mã chương trình Matlab được cho ở file **NVD10_MCQPSKber.m** (có trong Phụ lục 10A).

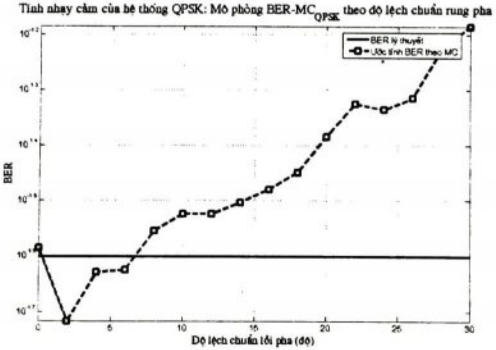
Kết quả chạy chương trình mô phỏng **NVD10_MCQPSKber.m** cho phép khảo và so sánh BER lý thuyết và mô phỏng theo SNR trong môi trường kênh AWGN được minh họa ở hình 10.7. Cho thấy kết quả mô phỏng là rất gần với kết quả AWGN lý thuyết.

Tiếp theo, ta nghiên cứu ảnh hưởng của rung pha (jitter) lên hiệu năng BER hệ thống. Quá trình lỗi pha được mô hình hóa là tạp âm Gausơ trắng. Mã chương trình Matlab mô phỏng tương ứng được ở file **NVD10_MCQPSKPhasejitter.m** trong Phụ lục 10A.



Hình 10.7: Các kết quả mô phỏng và lý thuyết cho một hệ thống QPSK hoạt động trong môi trường kênh AWGN

Kết quả mô phỏng tính nhạy cảm của hệ thống QPSK đối với rung pha (phase jitter) được minh họa ở hình 10.8. Như được mong đợi, BER tăng theo sự tăng của độ lệch chuẩn rung pha. Trong nhiều mô phỏng hệ thống, việc mô hình hóa rung pha là một quá trình tạp âm trắng là không phù hợp. Ta nên giải quyết vấn đề này theo cách, thiết kế bộ lọc FIR để thực thi hóa mật độ phổ công suất PSD của quá trình rung pha.

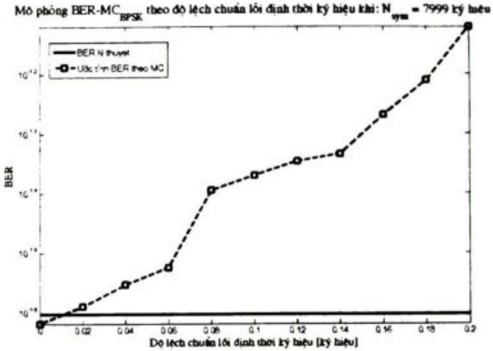


Hình 10.8: Mô phỏng minh họa tính nhạy cảm của hệ thống QPSK đối với rung pha

Chương trình mô phỏng cuối cùng là khảo sát mức độ nhạy cảm của BER đối với lỗi định thời ký hiệu. Chương trình Matlab tương ứng được cho ở file **NVD10_MCQPSKSymjitte.m** (có trong Phụ lục 10A) như sau.

Kết quả chạy chương trình mô phỏng **NVD10_MCQPSKSymjitte.m** được cho ở hình 10.9. Cũng như trong trường hợp rung pha, lỗi đồng bộ ký hiệu được mô hình hóa là quá trình

stochastic Gausơ trắng. Một lần nữa, nếu có tính nhớ ảnh hưởng lên quá trình rung ký hiệu, thì phải được mô hình hóa chính xác, ta có thể thiết kế một bộ lọc FIR để thực thi PSD.



Hình 10.9: Kết quả mô phỏng minh họa ảnh hưởng của rung ký hiệu

Ngoài ra, trong mô phỏng này các ký hiệu phát được tương quan chéo nhau (lưu ý việc sử dụng hàm **vxcorr**) nhằm đảm bảo các ký hiệu phát và thu được đồng chỉnh chính xác vì vậy BER được xác định chính xác. Trong những mô phỏng tiếp theo, bao gồm các hệ thống minh họa ở phần 10.3.3 và 10.3.4 dưới đây, ta dùng các kỹ thuật tương quan chéo để tính toán giá trị trễ và không cần phải mô phỏng riêng biệt để xác định tham số này. Ở đây ta dùng chương trình mô phỏng riêng biệt để minh họa mức độ nhạy cảm của các kết quả mô phỏng đối với các tham số quan trọng này. Mô phỏng này lại được dùng trong chương 16 khi ta xét lấy mẫu qua trọng.

10.3. Kỹ thuật bán phân tích

Như ta đã thấy, phương pháp mô phỏng Monte Carlo là phương pháp hoàn toàn tổng quát và có thể áp dụng cho hệ thống bất kỳ, trong đó các mô hình mô phỏng của các khối chức năng được định nghĩa, hoặc ít nhất là được xấp xỉ ở dạng một giải thuật (xử lý tín hiệu số DSP). Giá phải trả khi sử dụng các phương pháp Monte Carlo là thời gian chạy chương trình mô phỏng. Nếu các mô hình hệ thống và kênh bị phức tạp hóa và BER thấp thì yêu cầu thời gian chạy mô phỏng quá lâu vì vậy việc sử dụng các kỹ thuật Monte Carlo trở nên không thực tế nhưng vẫn là các mô phỏng quan trọng nhất.

Dưới đây, ta nhấn mạnh vào việc mô phỏng BER, vì BER là phép đo chung nhất để ước lượng hiệu năng hệ thống truyền thông số. Trong quá trình thực hiện mô phỏng, để ước tính BER thì thông tin được tập hợp để từ đó các mục khác được xác định. Các mục này bao gồm: dạng sóng, biểu đồ mắt, chòm sao tín hiệu, và ước tính mật độ phổ công suất PSD tại các điểm khác nhau của hệ thống.

May thay, có sự lựa chọn khác cho phương pháp Monte Carlo. Phương pháp mạnh mẽ nhất là phương pháp bán phân tích SA (***SemianAlytic***), trong đó sử dụng đồng thời kỹ thuật mô phỏng và phân tích để có được ước tính BER rất nhanh. Phương pháp mô phỏng bán giải tích

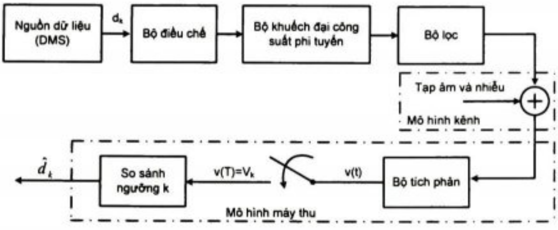
phù hợp với mọi kỹ thuật mô phỏng nhanh, nó cho phép tận dụng các kiến thức về giải tích để dung hòa với thời gian chạy mô phỏng.

Sơ đồ khối hệ thống đơn giản cho kỹ thuật mô phỏng SA khả dụng được minh họa ở hình 10.10. Ký hiệu phát thứ *k* là d_k và tương ứng ở đầu ra máy thu là $\hat{d}_k$. Ký hiệu phát được thu chính xác nếu $\hat{d}_k = d_k$ và tạo ra lỗi nếu $\hat{d}_k \neq d_k$. Từ lý thuyết truyền thông cơ bản cho thấy, đại lượng V_k là con số quyết định đối với ký hiệu phát thứ *k*, và máy thu thực hiện quyết định bằng cách so sánh giá trị của V_k với ngưỡng T.

Con số quyết định V_k được cho ở hình 10.10 là một hàm của ba thành phần:

$$V_k = f(S_k, D_k, N_k) \tag{10.2}$$

Trong đó, S_k thể hiện cho tín hiệu phát, D_k thể hiện cho méo hệ thống (ISI do lọc hoặc đa đường), N_k thể hiện cho các nhiễu loạn kênh (tạp âm và nhiễu). Khi ứng dụng SA, ta kết hợp S_k và D_k với nhau, nó được xác định bởi mô phỏng Monte Carlo, các ảnh hưởng của tạp âm N_k được xử lý theo cách giải tích. Kỹ thuật mô phỏng bán giải tích SA có thể ứng dụng cho hệ thống bất kỳ, trong đó hàm mật độ xác suất thành phần tạp âm của V_k được xác định theo cách giải tích. Trường hợp đơn giản, kênh tạp âm Gausơ cộng trong đó hệ thống được xem là tuyến tính so với điểm tạp âm được chèn vào (điểm mà ở đó con số quyết định V_k được xác định). Từ hình 10.10 cho thấy, nếu tạp âm kênh phân bố Gausơ, thì con số quyết định V_k là biến ngẫu nhiên Gausơ có trung bình được xác định bởi S_k và D_k. Vì vậy mô phỏng bán phân tích là kết hợp của mô phỏng Monte Carlo và phân tích.



Hình 10.10: Hệ thống để minh họa kỹ thuật mô phỏng bán giải tích SA

10.3.1. Khái niệm cơ bản

Ta xét hệ thống BPSK hoạt động trong môi trường kênh AWGN. Ta bỏ qua bộ lọc phát và nhận tín hiệu đáp ứng đầy đủ. Hàm mật độ xác suất pdf của V_k với điều kiện đã phát bit 1 hoặc 0 là phân bố Gausơ được minh họa ở hình 10.11.

Vì vậy, pdf của con số quyết định V_k với điều kiện đã phát $d_k = 1$ và $d_k = 0$ được cho bởi:

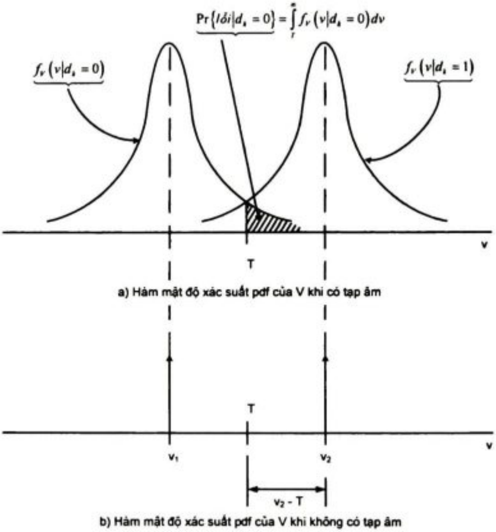
$$f_V(v|d_k = 0) = \frac{1}{\sqrt{2\pi}\sigma_v} e^{-\frac{(v-v_1)^2}{2\sigma_v^2}} \tag{10.3}$$

$$f_V(v|d_k = 1) = \frac{1}{\sqrt{2\pi}\sigma_v} e^{-\frac{(v-v_2)^2}{2\sigma_v^2}} \tag{10.4}$$

Trong đó v_1 và v_2 là các giá trị trung bình của biến ngẫu nhiên V_k với điều kiện đã phát $d_k = 0$ và $d_k = 1$, và với T là ngưỡng quyết định thì: *(i)* Xác suất lỗi với điều kiện đã phát $d_k = 0$; *(ii)* Xác suất lỗi với điều kiện đã phát $d_k = 1$ là:

$$\Pr\{\text{lỗi}|d_k = 0\} = \int_T^{\infty} f_V(v|d_k = 0)\,dv \tag{10.5}$$

$$\Pr\{\text{lỗi}|d_k = 1\} = \int_{-\infty}^{T} f_V(v|d_k = 1)\,dv \tag{10.6}$$



Hình 10.11: Quá trình quyết định nhị phân

Nếu việc phát ký hiệu $d_k = 0$ và $d_k = 1$ là đồng xác xuất, thì ngưỡng quyết định tối ưu là điểm tại đó hai hàm mật độ xác suất lỗi có điều kiện pdf bằng nhau (nghĩa là, **Pr(*lỗi/d*$_k$ *= 1*)** = Pr(*lỗi/d*$_k$ *= 0*)). Trong trường hợp này, các xác suất lỗi có điều kiện bằng nhau và xác suất lỗi tổng là

$$P_E = \underbrace{\underbrace{\Pr\{d_k = 0\}}_{\text{xác suất phát bit 0}} . \underbrace{\Pr\{1|d_k = 0\}}_{\substack{\text{xác suất quyết định bit 1 với} \\ \text{điều kiện bit 0 đã được phát đi}}} + \underbrace{\Pr\{d_k = 1\}}_{\text{xác suất phát bit 1}} . \underbrace{\Pr\{0|d_k = 1\}}_{\substack{\text{xác suất quyết định bit 0} \\ \text{với điều kiện bit 1 đã được phát đi}}}}_{\Pr\{d_k = 0\} \;=\; \Pr\{d_k = 1\} = \frac{1}{2}} \tag{10.7}$$

$$= \frac{1}{2}\Pr\{\text{lỗi}|d_k = 0\} + \frac{1}{2}\Pr\{\text{lỗi}|d_k = 1\}$$

Nó là:

$$\begin{aligned} P_E &= \Pr\{\text{lỗi}|d_k=0\} = \int_T^{\infty} f_V(v|d_k=0)dv \\ &= \Pr\{\text{lỗi}|d_k=1\} = \int_{-\infty}^{T} f_V(v|d_k=1)dv \end{aligned} \tag{10.8}$$

Xác suất đối với môi trường kênh AWGN thường được biểu diễn theo hàm Q, ta có:

$$P_E = Q\left(\frac{v_1}{\sigma_v}\right) \tag{10.9}$$

Trong đó hàm Q(.) Gausơ được định nghĩa như sau:

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^{\infty} e^{-\frac{t^2}{2}}dt \tag{10.10}$$

Lưu ý rằng, khi này các tín hiệu năng lượng bằng nhau, dẫn đến ngưỡng $T = 0$ và $v_2 = -v_1$. Vì vậy, khi xác định được v_1 và σ_n ta hoàn toàn xác định được BER hệ thống. Để xác định xác suất lỗi, ta chỉ cần triển khai một mô phỏng để ước tính v_1 và σ_n. Không cần đến kỹ thuật đếm lỗi Monte Carlo.

Giá trị của v_1 được xác định bằng cách mô phỏng không tạp âm. Nếu tạp âm kênh bị khử, thì hai hàm mật độ xác suất có điều kiện pdf được cho ở hình 10.11 (a) xẹp thành các hàm xung kim ($\sigma_n = 0$) như được thấy ở hình 10.11(b). Mỗi hàm xung kim đều có vùng diện tích bằng 1 và các vị trí của các hàm xung kim xác định v_1 và v_2.

Giá trị của σ_n được xác định bằng cách mô phỏng phần của hệ thống mà tạp âm đi qua. Với hệ thống đang được xét, mô hình hóa máy thu là bộ tách sóng tích phân-và-kết suất (*Integrate-And-Dump Detector*). Giả sử phần hệ thống này có hàm truyền đạt H(f). Nếu tạp âm trắng (tương quan là hàm Delta) có mật độ phổ công suất hai phía là $N_0/2$ được đặt vào máy thu bộ lọc thích hợp, thì phương sai của biến ngẫu nhiên V_k là:

$$\sigma_v = \frac{N_0}{2}\int_{-\infty}^{\infty}|H(f)|^2 df = N_0\int_0^{\infty}|H(f)|^2 df \tag{10.11}$$

Như đã thấy ở chương 7, độ rộng băng tần tương đương, cũng là độ rộng băng tần tạp âm tương đương của máy thu được định nghĩa là:

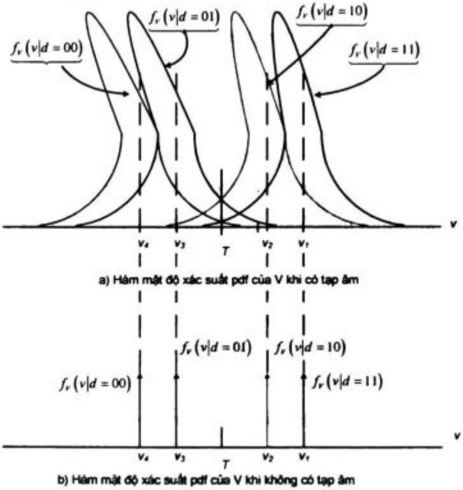
$$B_N = \int_0^{\infty}|H(f)|^2 df \tag{10.12}$$

Theo đó, xác suất lỗi được cho bởi :

$$P_E = Q\left(\frac{v_1}{\sqrt{N_0 B_N}}\right) \tag{10.13}$$

Lưu ý rằng, mặc dù hệ thống được xét trong môi trường kênh AWGN, nhưng bộ khuếch đại phi tuyến có thể ảnh hưởng đến hiệu năng hệ thống vì bộ khuếch đại phi tuyến sẽ ảnh hưởng lên dạng của các tín hiệu phát và làm ảnh hưởng đến giá trị của v_I.

Đến đây, ta xét ảnh hưởng của bộ lọc phát. Ảnh hưởng của bộ lọc phát làm tán thời các ký hiệu phát, nó gây ra ISI. Nếu độ dài nhớ của bộ lọc này là hai ký hiệu thì xác xuất lỗi truyền ký hiệu sẽ không chỉ phụ thuộc vào ký hiệu phát mà còn phụ thuộc vào ký hiệu được phát đi trước đó. Hậu quả, việc tính toán xác suất lỗi sẽ bao gồm *bốn* hàm mật độ xác suất có điều kiện chứ không phải là *hai* hàm mật độ xác suất có điều kiện như được cho ở hình 10.11. Điều này được minh họa ở hình 10.12 (a).



Hình 10.12: Các hàm mật độ xác suất pdf có điều kiện khi độ dài nhớ của bộ lọc phát là hai ký hiệu

Giống như trên, thực hiện mô phỏng không tạp âm sẽ cho ta các giá trị của v_1, v_2, v_3, v_4. Xác suất lỗi hệ thống khi này là:

$$P_E = \frac{1}{4}\sum_{i=1}^{4} Q\left(\frac{v_i}{\sigma_v}\right) \tag{10.14}$$

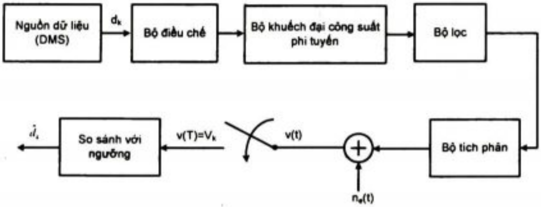
10.3.2. Nguồn tạp âm tương đương

Trong quá trình ứng dụng kỹ thuật bán giải tích SA ta đã dùng ý tưởng về một nguồn tạp âm tương đương. Ta đã biết rằng, con số quyết định V_k là một hàm của ba thành phần, $V_k = f(S_k, D_k, N_k)$, trong đó S_k thể hiện cho tín hiệu; D_k thể hiện cho méo hệ thống (ISI) và N_k thể hiện cho tạp âm. Ảnh hưởng của S_k và D_k được xác định bởi mô phỏng Monte Carlo, và ảnh

hưởng của tạp âm N_k được xử lý theo cách giải tích. Nếu thực hiện mô phỏng phi tạp âm, thì con số thích hợp (ký hiệu là $V_{k,nf}$) sẽ chỉ là một hàm của S_k và D_k. Để con số này được cộng với một biến ngẫu nhiên N_k có phương sai được định nghĩa bởi (10.11), thì:

$$V_k = V_{k,nf} + N_k \tag{10.15}$$

Biến ngẫu nhiên N_k được xem là lấy mẫu từ một nguồn tạp âm $n_e(t)$ như được thấy ở hình 10.13. Nguồn tạp âm tương đương này gồm các ảnh hưởng của tạp âm nhiệt, nhiễu, và các suy thoái kênh, chúng được phản ánh tại đầu ra bộ tích phân. Nếu tập âm kênh là trắng, thì đáp ứng xung (hay hàm truyền đạt được định nghĩa bởi (10.11)) được dùng để truyền tạp âm kênh tới đầu ra bộ lấy tích phân.



Hình 10.13: Nguồn tạp âm tương đương đối với mô phỏng bán giải tích SA

10.3.3. Ước tính BER bán giải tích cho PSK

Tại đây, ta khai triển giải thuật để xác định BER cho hệ thống BPSK bằng mô phỏng bán giải tích, và ta thực hiện theo cách để có thể dễ dàng mở rộng cho QPSK. Xét chòm sao tín hiệu được minh họa ở hình 10.14, nó cho thấy các điểm tín hiệu phát S_1 và S_2 tương ứng với các vùng quyết định D_1 và D_2. Một quyết định chính xác ở máy thu, nếu S_i được phát đi và tín hiệu thu rơi vào vùng D_i; nếu ngược lại thì xảy ra lỗi. Trong hình 10.14, ta coi rằng S_1 được phát và thu được $\tilde{S}_1$. Như đã đề cập, do ISI, méo phi tuyến, suy thoái kênh... làm cho $S_1 \neq \tilde{S}_1$. Sự khác nhau giữa S_1 và $\tilde{S}_1$ được ký hiệu là d_x. Xác suất lỗi với điều kiện S_1 đã được phát đi là:

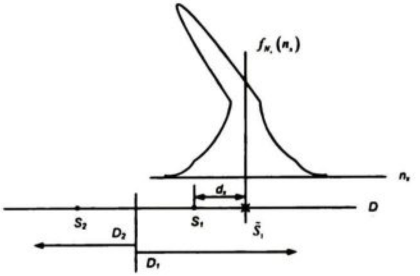
$$\Pr\{\text{lỗi}|S_1\} = \int_{\tilde{S}_1+n\notin D_1} \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(n-\tilde{S}_1)^2}{2\sigma_n^2}} dn \tag{10.16}$$

$$\Leftrightarrow \Pr\{\text{lỗi}|S_1\} = \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(n-\tilde{S}_1)^2}{2\sigma_n^2}} dn \tag{10.17}$$

Dưới dạng hàm Q Gausơ, phương trình trên trở thành:

$$\Pr\{\text{lỗi}|S_1\} = Q\left(\frac{\tilde{S}_1}{\sigma_n}\right) \tag{10.18}$$

Như vậy, nếu biết $\tilde{S}_1$ (được xác định bằng cách dùng mô phỏng Monte Carlo) và σ_n cho phép ta xác định được BER có điều kiện. Để xác định σ_n, cần phải xác định giá trị của B_N từ đáp ứng xung được mô phỏng *h[n]*.



Hình 10.14: Ước tính BER bán phân tích cho PSK

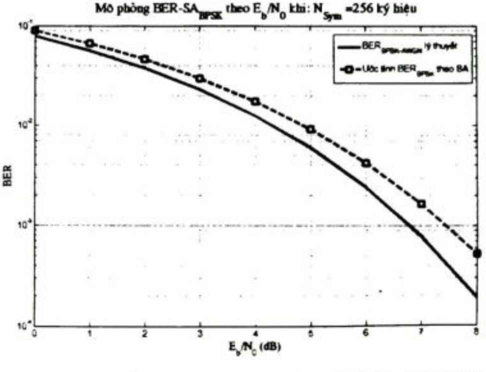
Giả sử S_k là bit phát thứ *k* trong chuỗi *N* bít được mô phỏng. Với mỗi giá trị của *k*, $1 \le k \le N$, thì S_k sẽ là S_1 hoặc S_2. BER có điều kiện là:

$$\Pr\{\text{lỗi}|\text{S}_k\} = Q\left(\frac{\tilde{\text{S}}_k}{\sigma_n}\right) \tag{10.19}$$

BER tổng đạt được bằng cách lấy trung bình trên toàn bộ chuỗi N bit như sau:

$$P_E = \frac{1}{N}\sum_{k=1}^{N} Q\left(\frac{\tilde{\text{S}}_k}{\sigma_n}\right) \tag{10.20}$$

Chương trình mô phỏng hiệu năng BER hệ thống BPSK theo phương pháp bán phân tích



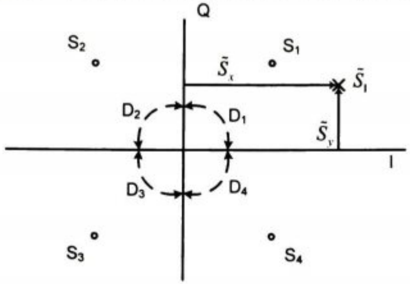
Hình 10.15: Ước tính BER bán phân tích đối với BPSK

Mã chương trình Matlab thực hiện mô phỏng bán giải tích cho hệ thống PSK được cho ở file **NVD10_PSKSA.m** (có trong Phụ lục 10A). Phương pháp luận được dùng để trình bày giống như trên. Do tính đối xứng, nên tín hiệu thu bị quay theo các giá trị dương. Độ rộng băng tần của bộ lọc phát gây ra ISI bằng với tốc độ bit. Thấy rõ từ hình 10.15, ảnh hưởng của ISI lên hiệu năng BER.

10.3.4. Ước tính BER bán phân tích cho QPSK

Ta xét bộ mô phỏng bán phân tích cho xác suất lỗi *ký hiệu* P_S trong hệ thống QPSK. Vì một chòm sao tín hiệu QPSK có 4 điểm tín hiệu chứ không phải là hai, và vì không gian tín hiệu có hai chiều chứ không phải là một như trường hợp BPSK, nên bộ ước tính bán phân tích cho QPSK khác với BPSK, trong đó phải thêm một chiều nữa để tạo kênh vuông pha.

Ta xét chòm sao tín hiệu được minh họa ở hình 10.16. Các điểm tín hiệu phát S_i, $i = 1$, 2, 3, 4 tương ứng các vùng quyết định D_i, $i = 1,2,3,4$. Giống như phần trước, quyết định đúng ở máy thu nếu S_i được phát đi và tín hiệu thu rơi vào vùng quyết định D_i; nếu ngược lại thì xảy ra lỗi. Cụ thể trong hình 10.16, ta giả sử S_1 được phát đi và tín hiệu thu không có tạp âm được ký hiệu là $\tilde{S}_1$. Hậu quả của ISI và méo làm cho $\tilde{S}_1 \neq S_1$. Nó là $\tilde{S}_1$ chứ không phải là S_1 (được xác định bởi mô phỏng bán phân tích) bởi vì mô phỏng sẽ làm sáng tỏ các ảnh hưởng của ISI chứ không làm sáng tỏ các ảnh hưởng của tạp âm. Các thành phần đồng pha và vuông pha của $\tilde{S}_1$ được ký hiệu là $\tilde{S}_x$ và $\tilde{S}_y$ trong đó $\tilde{S}_x = \mathrm{Re}\left(\tilde{S}_1\right)$ và $\tilde{S}_y = \mathrm{Im}\left(\tilde{S}_1\right)$. Một khi xét đến tạp âm bằng cách cộng n_x và n_y với $\tilde{S}_x$ và $\tilde{S}_y$, thì một quyết định đúng được thực hiện với điều kiện S_1 đã được phát đi nếu nếu $\left(\tilde{S}_x + n_x, \tilde{S}_y + n_y\right) \in D_1$. Thực hiện quyết định sai nếu $\left(\tilde{S}_x + n_x, \tilde{S}_y + n_y\right) \notin D_1$. Cần phải lưu ý rằng, do ta đang triển khai một bộ ước tính bán phân tích, nên ảnh hưởng của tạp âm được xử lý theo cách phân tích và không xuất hiện trong hình 10.16.



Hình 10.16: Ước tính BER bán phân tích cho QPSK

Vấn đề ở chỗ, xác định các thành phần tạp âm n_x và n_x mà sẽ dẫn đến một lỗi cho điểm tín hiệu thu (không có tạp âm) trong không gian tín hiệu $\tilde{S}_1$. Vấn đề này rất giống với trường

hợp BPSK đã xét nhưng có sự khác biệt căn bản là không gian tín hiệu. Ta giả thiết rằng, các thành phần tạp âm cộng đồng pha và vuông pha là không tương quan nhau và đều là phân bố Gausơ. Theo đó, giả sử S_I được phát đi và thu được $\tilde{S}_1$, thì xảy ra một lỗi nếu:

$$\Pr\{\text{lỗi}|S_1\} = \iint_{(\tilde{S}_x+n_x,\tilde{S}_y+n_y)\in D_1} \frac{1}{2\pi\sigma_n\sigma_n}.e^{\left(-\frac{(n_x-\tilde{S}_x)^2}{2\sigma_n^2}-\frac{(n_y-\tilde{S}_y)^2}{2\sigma_n^2}\right)}dn_x dn_y \qquad (10.21)$$

Trong đó n_x và n_y là các thành phần tạp âm đồng pha và vông pha và σ_n thể hiện cho phương sai. Để đơn giản hóa ký hiệu, ta đặt:

$$f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right) = \frac{1}{\sqrt{2\pi}\sigma_n}e^{-\frac{(n_x-\tilde{S}_x)^2}{2\sigma_n^2}} \qquad (10.22)$$

$$f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right) = \frac{1}{\sqrt{2\pi}\sigma_n}e^{-\frac{(n_y-\tilde{S}_y)^2}{2\sigma_n^2}} \qquad (10.23)$$

Khi này (10.21) trở thành:

$$\Pr\{\text{lỗi}|S_1\} = \iint_{(\tilde{S}_x+n_x,\tilde{S}_y+n_y)\in D_1} f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right).f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right)dn_x dn_y \qquad (10.24)$$

Xác suất lỗi này có thể được giới hạn bởi biểu thức:

$$\begin{aligned}\Pr\{\text{lỗi}|S_1\} < &\iint_{(\tilde{S}_x+n_x,\tilde{S}_y+n_y)\in(D_2\cup D_3)} f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right).f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right)dn_x dn_y \\ &+ \iint_{(\tilde{S}_x+n_x,\tilde{S}_y+n_y)\in(D_3\cup D_4)} f_{N_x}(n_x).f_{N_y}(n_y)dn_x dn_y\end{aligned} \qquad (10.25)$$

Trong đó xuất hiện ranh giới vì vùng quyết định D_3 xảy ra hai lần trong (10.25). Từ định nghĩa về các vùng quyết định ta có thể viết:

$$\begin{aligned}\Pr\{\text{lỗi}|S_1\} < &\int_{-\infty}^{0} f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right)dx \int_{-\infty}^{\infty} f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right)dy \\ &+ \int_{-\infty}^{\infty} f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right)dn_x \int_{-\infty}^{0} f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right)dn_y\end{aligned} \qquad (10.26)$$

Thừa nhận rằng, hai trong bốn tích phân trong (10.26) là bằng 1, nhận được:

$$\Pr\{\text{lỗi}|S_1\} < \int_{-\infty}^{0} f_{N_x}\left(n_x\middle|\tilde{S}_x,\sigma_n\right)dn_x + \int_{-\infty}^{0} f_{N_y}\left(n_y\middle|\tilde{S}_y,\sigma_n\right)dn_y \qquad (10.27)$$

Thế (10.22) và (10.23) vào biểu thức trên và dùng các định nghĩa về $\tilde{S}_x$ và $\tilde{S}_y$ nhận được giới hạn cho xác suất lỗi có điều kiện. Giới hạn xác suất lỗi có điều kiện này là:

$$\Pr\{\text{lỗi}|S_1\} < Q\left(\frac{\text{Re}(\tilde{S}_1)}{\sigma_n}\right) + Q\left(\frac{\text{Im}(\tilde{S}_1)}{\sigma_n}\right) \tag{10.28}$$

Trong đó, hàm Q(.) là hàm Q Gausơ. Do tính đối xứng xác suất có điều kiện là như nhau đối với bốn ký hiệu phát có thể có.

Giống như trường hợp PSK ta đã giả thiết S_k là ký hiệu phát thứ k trong chuỗi N ký hiệu được mô phỏng. Với mỗi giá trị của k, $1 \le k \le N$, thì S_k sẽ là S_1, S_2, S_3, hoặc S_4. Ranh giới cho tỉ số lỗi ký hiệu có điều kiện là, từ (10.28):

$$\Pr\{\text{lỗi}|S_k\} < Q\left(\frac{\text{Re}(\tilde{S}_k)}{\sigma_n}\right) + Q\left(\frac{\text{Im}(\tilde{S}_k)}{\sigma_n}\right) \tag{10.29}$$

Tỉ số lỗi ký hiệu tổng đạt được bằng cách lấy trung bình xác suất lỗi ký hiệu có điều kiện trên toàn bộ chuỗi N ký hiệu được cho bởi:

$$\Pr\{\text{lỗi}|S_1\} < \frac{1}{N}\sum_{k=1}^{N}\left[Q\left(\frac{\text{Re}(\tilde{S}_k)}{\sigma_n}\right) + Q\left(\frac{\text{Im}(\tilde{S}_k)}{\sigma_n}\right)\right] \tag{10.30}$$

Tỉ số lỗi bít $P_E = \frac{P_S}{2}$. Lưu ý rằng, trong trường hợp PSK ta đạt được một nghiệm chính xác, trong khi đó với QPSK ta có một giới hạn (ranh giới). Kỹ thuật được dùng ở đây để phát triển bộ ước tính bán giải tích dễ dàng được mở rộng cho MPSK và QAM.

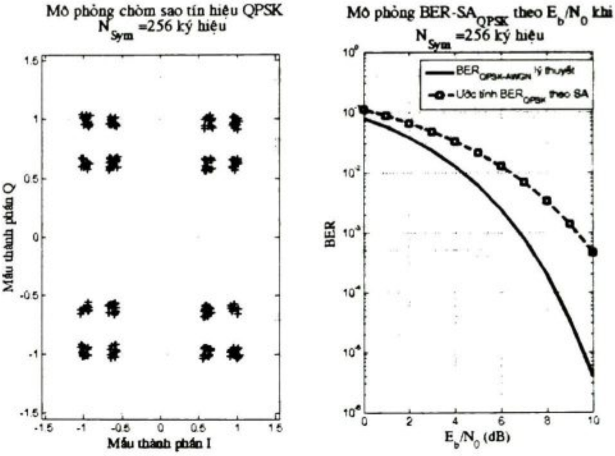
Bộ ước tính được triển khai ở đây sẽ được dùng thông suốt các phần còn lại của tài liệu nhằm ước lượng hiệu năng của một số hệ thống. Quy tụ lại sẽ là các ví dụ để minh họa ảnh hưởng của đa đường và phadinh trong hệ thống vô tuyến và ảnh hưởng của méo phi tuyến trong hệ thống truyền thông vệ tinh ghép kênh theo tần số.

Chương trình mô phỏng hiệu năng BER hệ thống QPSK theo phương pháp bán giải tích

Mã chương trình Matlab thực hiện mô phỏng bán giải tích hệ thống QPSK được cho ở file **NVD10_QPSKSA.m** dưới đây (có trong phụ lục 10A). Thực hiện mô phỏng cho phép nghiên cứu các ảnh hưởng của ISI do lọc phát gây ra. Độ rộng băng tần được đặt bằng tốc độ ký hiệu (là một nửa tốc độ bit nghĩa là $BW = r_b/2$). Vì chòm sao tín hiệu là đối xứng, nên tất cả các điểm tín hiệu thu bị quay đến cung phần tư thứ nhất như đã được đề cập ở trên.

Thực hiện mô phỏng, ta được chòm sao tín hiệu và BER được minh họa ở hình 10.17. Lưu ý rằng, chòm sao tín hiệu thu không còn chứa 4 điểm như trường hợp QPSK lý tưởng nữa mà bây giờ nó gồm 16 điểm. Để làm rõ vấn đề này, ta giả sử rằng điểm tín hiệu thu trong cung phần tư thứ nhất trình bày cho bít dữ liệu 00 và do tính có nhớ của hệ thống gây ra ISI là hai ký hiệu (các ký hiệu phát hiện tại và ký hiệu phát trước đó). Hậu quả là tạo ra 4 điểm tín hiệu từ việc truyền cặp bit 00. Bốn điểm tín hiệu này tương ứng với 00|00, 00|01, 00|10, 00|11, trong đó thanh thẳng đứng phác họa cho ký hiệu hiện thời và các ký hiệu được phát trước đó. Cũng cần lưu ý rằng mỗi điểm trong số 4 điểm ở cung phần tư thứ nhất được bao gồm các điểm mà hơi bị

tán xạ. Hiện tượng tán xạ này do hệ thống thể hiện tính chất nhớ với độ lâu lớn hơn hai ký hiệu, mặc dù ảnh hưởng của tính có nhớ này là nhỏ. Mặt phẳng phía bên phải của hình 10.17 minh họa cho BER của hệ thống khi có lọc phát. Kết quả mô phỏng cho kênh AWGN cũng được đưa ra nhằm so sánh và tham khảo. Thấy rõ BER tăng lên do ISI.



Hình 10.17: Kết quả mô phỏng bán giải tích QPSK

10.3.5. Chọn chuỗi dữ liệu

Khi ứng dụng kỹ thuật bán giải tích vào hệ thống có nhớ, một điều rất quan trọng là dùng nguồn dữ liệu để tạo ra các chuỗi thể hiện tất cả các kết hợp có thể có của các ký hiệu dữ liệu đối với độ dài nhớ cho trước của hệ thống. Ví dụ, nếu độ dài nhớ là 3 (gồm ký hiệu hiện tại cộng với 2 ký hiệu trước đó), thì xác suất lỗi ký hiệu được cho bởi:

$$P_S < \frac{1}{N}\sum_{k=1}^{N} \Pr\left\{\text{lỗi}\middle|S_k, S_{k-1}, S_{k-2}\right\} \qquad (10.31)$$

Ở dạng tổng quát, xác suất lỗi sẽ khác nhau đối với mỗi chuỗi (S_k, S_{k-1}, S_{k-2}). Theo đó, tất cả các kết hợp của (S_k, S_{k-1}, S_{k-2}) phải xảy ra bằng số lần để làm sáng tỏ ảnh hưởng nhớ. Nói chung, nếu một hệ thống nhị phân thể hiện tính có nhớ kéo dài tới B ký hiệu, thì tất cả các chuỗi nhị phân độ dài N nên được tạo ra bởi nguồn dữ liệu bằng số lần mô phỏng (mỗi lần mô phỏng cho một tổ hợp của kết hợp đó). Với hệ thống nhị phân có 2^N chuỗi độ dài N. Có ba cách thực hiện vấn đề này, hoặc ít nhất là lấy xấp xỉ như sau:

1. Nếu N đủ lớn, một khi muốn dùng chuỗi PN làm nguồn dữ liệu. Như đã được đề cập ở chương 7, số chuỗi được tạo ra sẽ không phải là 2^N như mong đợi mà sẽ là $L = 2^N - 1$ vì sẽ không bao giờ xuất hiện chuỗi N số 0 liên tiếp. Kết quả sẽ là một chuỗi không cân bằng có $\frac{L}{2}$

số 1 và $\frac{L}{2}-1$ số 0. Nếu N lớn, thì ảnh hưởng này có thể bỏ qua được. Lưu ý rằng một khi được tự do lựa chọn N lớn hơn độ dài tính có nhớ của hệ thống để giảm nhẹ ảnh hưởng này. Tuy nhiên, chọn N lớn hơn mức cần thiết dẫn đến thời gian thực hiện mô phỏng lâu hơn.

2. Nếu mốn có chuỗi hoàn toàn cân bằng, thì có thể dùng chuỗi deBrujjn. Như đã được đề cập vắn tắt ở chương 7, chuỗi deBrujjn được tạo ra bằng cách thêm số 0 vào vị trí N-1 số 0 trong đầu ra của bộ tạo chuỗi PN.

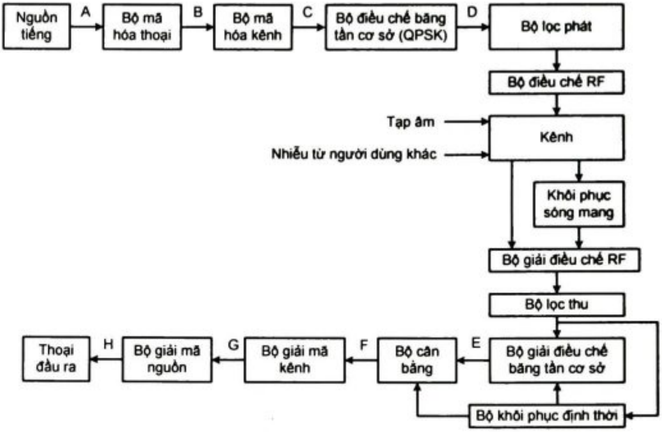
3. Tất nhiên, khi muốn đơn giản hóa việc thực hiện chuỗi bán giải tích thì nên dùng dữ liệu ngẫu nhiên. Nếu chuỗi này đủ dài, thì mọi kết hợp ký hiệu dữ liệu sẽ xuất hiện xấp xỉ bằng số lần. Đây là giải pháp đã được dùng trong các hai trường hợp mô phỏng cuối cùng.

Chương 11

PHƯƠNG PHÁP LUẬN MÔ PHỎNG HỆ THỐNG VÔ TUYẾN

11.1. Mở đầu

Trong chương này ta minh hoạ các khía cạnh của phương pháp luận nhằm ước tính hiệu năng hệ thống truyền thông vô tuyến làm việc trong môi trường kênh *pha đinh chậm*. Ta bắt đầu từ sơ đồ khối hệ thống được cho ở hình 11.1.



Hình 11.1: Mô hình mô phỏng mức hệ thống

Giả sử thiết kế hệ thống gần như hoàn thiện và các khía cạnh sau đây của thiết kế đã được hoàn tất:

1. Tín hiệu thoại là nguồn tín hiệu được mã hoá bởi mã hoá dự đoán tuyến tính tạo ra tốc độ bit đầu ra là 9600 bit/s.

2. Mã hóa kiểm soát lỗi dùng bộ lập mã chập tỷ lệ 1/3 và giải mã quyết định cứng (hoặc giải mã quyết định mềm với 8 mức lượng tử).

3. Dùng các bộ lọc loại SQRC 50%.

4. Dùng bộ cân bằng bình phương trung bình tuyến tính (LMS) đồng bộ 9 nhánh.

5. Điều chế QPSK với giải điều chế nhất quán tại máy thu.

Kênh được giả định là kênh đa đường "2 tia" với pha đinh chậm (chậm so với tốc độ ký hiệu sao cho kênh được xem là tựa tĩnh). Theo đó, quan hệ vào/ra của kênh được cho bởi:

$$\tilde{y}(t) = \tilde{a}_1(t)\tilde{x}(t-\tau_1(t)) - \tilde{a}_2(t)\tilde{x}(t-\tau_2(t)) \tag{11.1}$$

Trong đó $\tilde{x}(t)$ và $\tilde{y}(t)$ là các đầu vào/ra phức (thông thấp), $\tilde{a}_1(t)$ và $\tilde{a}_2(t)$ trình bày suy hao phức của 2 thành phần đa đường có trễ $\tau_1(t)$ và $\tau_2(t)$. Các suy hao phức được mô hình hóa là các quá trình dừng độc lập và độ rộng băng thông (tốc độ thay đổi) của các quá trình này được coi là nhỏ so với tốc độ ký hiệu. Các suy hao phức được mô hình hóa bởi hai quá trình Gausơ phức độc lập (*đường bao Rayleigh*) và *trễ được coi là phân bố đều* (kênh này sẽ được triển khai chi tiết ở chương 14).

Các đặc tính kênh thay đổi ngẫu nhiên là hàm của thời gian. Vì vậy, công suất tín hiệu thu và mức độ gây méo tín hiệu do kênh sẽ ảnh hưởng lên hiệu năng hệ thống, cũng sẽ thay đổi theo thời gian. Khi suy hao tín hiệu và méo là nhỏ, thì hiệu năng hệ thống sẽ rất tốt ngược lại thì hiệu năng hệ thống sẽ giảm đáng kể. Số đo hiệu năng toàn bộ trong hệ thống là chất lượng thoại đầu ra.

Mục đích của mô phỏng là ước lượng hiệu năng hệ thống khi được đo bởi số đo chất lượng thoại V, nó là hàm của E_b/N_0 và cần phải tính toán giá trị E_b/N_0 để duy trì xác suất ngừng hoạt động nhỏ hơn 2% tại mức ngưỡng phẩm chất chất lượng thoại là 3. Tại đây trình bày chi tiết toàn bộ giải pháp được dùng để ước tính xác suất ngừng hoạt động là hàm của E_b/N_0.

11.2. Tốc độ lấy mẫu và đơn giản hoá mức hệ thống

Giả định pha đinh chậm dẫn đến đơn giản hoá mô hình mô phỏng khi ước tính hiệu năng.

1. Đồng bộ hóa: Do pha đinh chậm, nên các phân hệ khôi phục pha và định thời có thể thiết lập chuẩn pha và định thời gần như lý tưởng. Vì vậy, khi ước tính hiệu năng có thể bỏ qua các phân hệ này trong mô hình mô phỏng. Khi này, coi đồng bộ hóa là lý tưởng.

2. Kênh tĩnh: Giả định pha đinh chậm dẫn đến kênh được coi là *tựa tĩnh* và dùng những chớp ảnh của kênh để ước tính hiệu năng. Tại đây, mô hình kênh đơn giản là:

$$\tilde{y}(t) = \tilde{a}_1\tilde{x}(t-\tau_1) - \tilde{a}_2\tilde{x}(t-\tau_2) \tag{11.2}$$

Trong đó $\tilde{a}_1$, $\tilde{a}_2, \tau_1$ và τ_2 là các biến ngẫu nhiên có giá trị không đổi trong mỗi lần mô phỏng ước tính hiệu năng. Thực tế thường chuẩn hóa $\tau_1 = 0$ và $\tilde{a}_1 = 1$ dẫn đến quan hệ vào/ra là:

$$\tilde{y}(t) = \tilde{x}(t) - \tilde{a}\tilde{x}(t-\tau) \tag{11.3}$$

Hàm truyền đạt kênh là:

$$H_c(f) = 1 - \tilde{a}e^{-j2\pi f\tau} \tag{11.4}$$

Trong mô hình này, kênh được đặc trưng bởi 2 biến ngẫu nhiên $\tilde{a}$ và τ trong đó $\tilde{a}$ có hàm pdf phân bố Rayleigh và τ có hàm pdf phân bố đồng đều.

3. Bộ điều chế và bộ giải điều chế RF: Giả định hai khối này chỉ thực hiện dịch tần số một cách lý tưởng, vì vậy chúng được loại bỏ ra khỏi mô hình mô phỏng. Vì vậy, toàn bộ hệ thống được mô phỏng dùng các biểu diễn tương đương thông thấp phức.

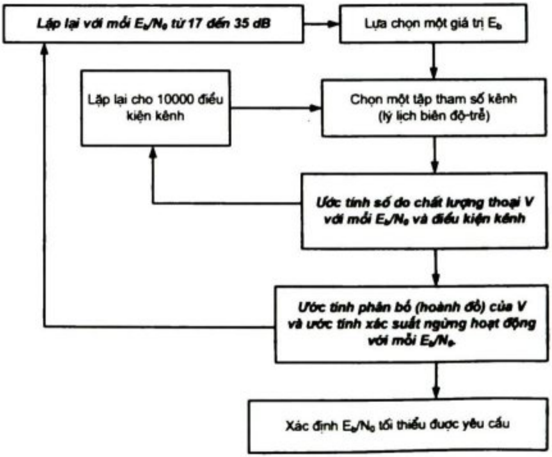
Tốc độ lấy mẫu

Tốc độ lấy mẫu tổng thể là tham số mô phỏng quan trọng phải được thiết lập khi bắt đầu. Nguồn tín hiệu thoại, bộ mã nguồn, khối lập mã kiểm soát lỗi ở phía phát, bộ giải mã kiểm soát

lỗi, khối giải mã nguồn làm việc trên các chuỗi ký hiệu, chúng nên được mô phỏng tại tốc độ một mẫu trên ký hiệu (nghĩa là, chúng được xử lý tại tốc độ ký hiệu hoặc bit phù hợp). Từ đầu ra bộ điều chế QPSK tới đầu ra bộ cân bằng, ta sẽ xử lý dạng sóng, vì vậy tín hiệu và các thành phần trong nhóm này, chúng nên được mô phỏng tại tốc độ lấy mẫu phù hợp với băng thông tín hiệu và thành phần. Vì không có các thành phần phi tuyến thay đổi theo thời gian LTV trong hệ thống, nên không cần xét đến hiện tượng mở rộng băng thông. Theo đó, không cần phải xét lấy mẫu đa tốc độ bởi lẽ chúng không liên quan đến các tín hiệu có các độ rộng băng tần khác nhau. Tốc độ lấy mẫu cho phần "*tương tự*" của hệ thống được thiết lập gấp 16 lần băng thông tín hiệu QPSK. Điều này có thể cắt băng thông của bộ lọc Cosin tăng, nó bằng 0,75 lần tốc độ ký hiệu (0,5R+50% của (0,5R) = 0.75R). Với tốc độ ký hiệu QPSK, R = (9600x3)/2 = 14.400 ký hiệu/giây, thì ta dùng tốc độ lấy mẫu 16x0,75x14.400 = 172.800 mẫu/giây tương đương với 12 mẫu/ký hiệu QPSK.

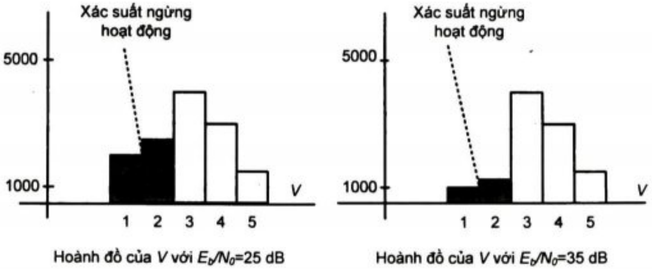
11.3. Phương pháp luận toàn diện

Khi ước tính xác suất ngừng hoạt động, ta cần phải xác định khoảng thời gian mà các điều kiện kênh làm giảm hiệu năng hệ thống dưới mức ngưỡng. Vì các tham số kênh là các biến ngẫu nhiên, nên ta dùng giải pháp Monte Carlo để xác định xác suất ngừng hoạt động. Giải pháp Monte Carlo bao gồm việc lấy ra các số ngẫu nhiên từ phân bố của các tham số kênh $\tilde{a}$ và τ, và tính toán hiệu năng hệ thống cho mỗi cặp giá trị của $\tilde{a}$ và τ. Ước tính xác suất ngừng hoạt động là phần trăm của các kênh được mô phỏng tạo ra số đo hiệu năng dưới mức chấp nhận (ngưỡng). Lưu ý rằng, phần mô phỏng Monte Carlo này là khác với mô phỏng Monte Carlo ước tính hiệu năng cho mỗi điều kiện kênh. Mô phỏng Monte Carlo để ước tính hiệu năng gồm việc tạo các giá trị mẫu của một hoặc nhiều quá trình ngẫu nhiên mà nó thể hiện cho các tín hiệu và tạp âm.



Hình 11.2: Lưu đồ ước tính xác suất ngừng hoạt động

Lưu đồ ước tính xác suất ngừng hoạt động được cho ở hình 11.2. Để ước tính xác suất ngừng hoạt động, cần phải tính hiệu năng hệ thống khi được đánh giá bởi số đo chất lượng thoại đầu ra cho mỗi giá trị của E_b/N_0 với hàng ngàn chớp ảnh của kênh. Ta thường định nghĩa kênh theo $\tilde{a}$ và τ (*lý lịch biên độ-trễ*). Một số điều kiện kênh sẽ gây ra suy thoái đáng kể hiệu năng dẫn đến số đo chất lượng thoại dưới mức ngưỡng. *Xác suất ngừng hoạt động ứng với giá trị E_b/N_0 cho trước được ước tính bằng tỉ số các kênh được mô phỏng (tạo ra số đo chất lượng thoại thấp hơn ngưỡng) trên tổng số các kênh được mô phỏng.* Khi E_b/N_0 tăng thì sẽ giảm xác suất ngừng hoạt động. Bằng cách lặp lại các mô phỏng cho các giá trị khác nhau của E_b/N_0 ta có thể tìm giá trị của E_b/N_0 (tối thiểu) mà đảm bảo xác suất ngừng hoạt động nhỏ hơn 2%. Ta giả sử rằng, dải giá trị của E_b/N_0 được xét là 17 đến 35 dB với bước nhảy 2 dB, và cũng giả thiết rằng với mỗi giá trị của E_b/N_0 ta mô phỏng 10000 điều kiện kênh để được phân bố (hoành đồ) của V và xác suất ngừng hoạt động. Các kết quả điển hình được minh hoạ trong hình 11.3. Thấy rõ từ hoành đồ, xác suất ngừng hoạt động với $E_b/N_0 = 35$ dB là nhỏ hơn so với trường hợp $E_b/N_0 = 25$ dB.



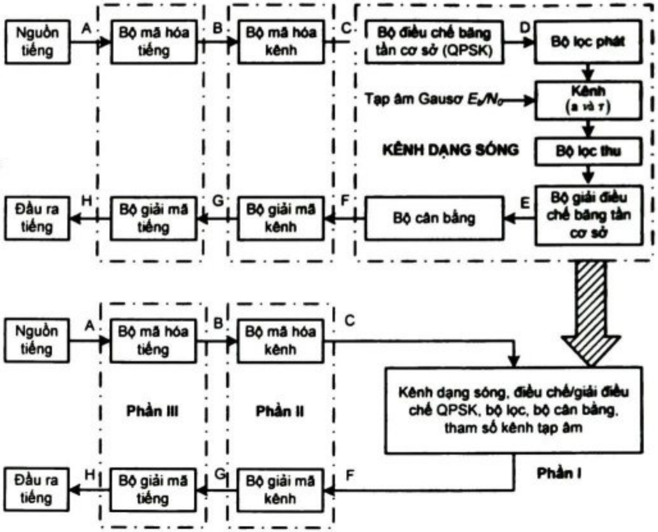
Hình 11.3: Hoành đồ điển hình trong tính toán xác suất ngừng hoạt động

Ước tính xác suất lỗi kênh dạng sóng

Khi cho trước điều kiện kênh và E_b/N_0 ta có thể ước tính số đo chất lượng thoại dùng giải pháp Monte Carlo ép thô trong đó ta dùng mẫu thoại và số thoại làm đầu vào, ghi âm lại đầu ra bộ giải mã thoại và kiểm tra bằng thính giác người. Giải pháp giả thực tế này, không khả thi để lặp lại cho hàng ngàn điều kiện kênh và nhiều giá trị E_b/N_0, thậm chí ta có các nguồn máy tính để làm mô phỏng, nó cũng yêu cầu mỗi người nghe cho điểm hàng ngàn đoạn thoại.

Giải pháp tốt hơn là chia bài toán thành các phần và mô phỏng các phần riêng biệt. Để tiến đến một sơ đồ phân chia hiệu quả, ta xét ảnh hưởng của các phần khác nhau lên toàn bộ hiệu năng ở dạng số đo chất lượng thoại. Theo hình 11.4, phần "*xử lý*" dạng sóng của hệ thống tiếp nhận chuỗi nhị phân tại điểm C và tạo ra chuỗi nhị phân cho giải mã quyết định cứng (hoặc các giá trị lượng tử khi giải mã quyết định mềm) tại điểm F (khối kênh dạng sóng-xem hình 11.4). Xác suất lỗi q (hoặc các xác suất chuyển đổi trạng thái đối với giải mã quyết định mềm) phụ thuộc vào các tham số của kênh và E_b/N_0. Xác suất lỗi này (hoặc tập các xác suất chuyển dịch đối với giải mã quyết định mềm) có thể được ước tính bằng kỹ thuật Monte Carlo, hoặc kỹ

thuật bán phân tích với các chuỗi nhị phân ngẫu nhiên làm đầu vào. Không cần thiết điều khiển phần mô phỏng này bằng các bit thoại mã hoá.



Hình 11.4: Phân chia hóa và đơn giản hóa mô hình mô phỏng

Ánh xạ xác suất lỗi kênh dạng sóng thành xác suất lỗi mã hóa

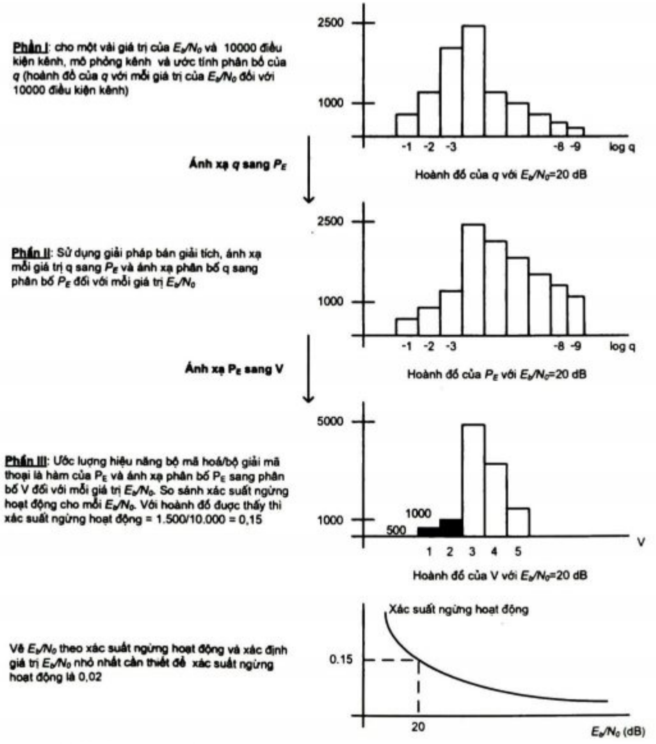
Đoạn tiếp theo của hệ thống gồm bộ lập mã kiểm soát lỗi và bộ giải mã, nó tiếp nhận chuỗi số nhị phân tại điểm B và tạo ra chuỗi số nhị phân tại điểm G. Xác suất lỗi giữa điểm B và G hoàn toàn là một hàm xác xuất lỗi q trong môi trường kênh dạng sóng. Thực ra, vì các lỗi trong kênh dạng sóng là kết quả của tạp âm Gausơ, trắng, cộng AWGN, nên ta coi mẫu lỗi là một chuỗi độc lập thuộc về ước lượng xác suất lỗi bít mã hoá giữa điểm B và G, khi này kênh dạng sóng được thay bằng bộ tạo số ngẫu nhiên nhị phân để tạo ra các bit 1 và 0 với xác suất q và 1-q, trong đó 1 thể hiện cho lỗi truyền dẫn trong kênh dạng sóng. Xác suất lỗi mã hoá P_E có thể được ước lượng nhờ mô phỏng Monte Carlo trong đó đầu vào bộ mã hoá là chuỗi nhị phân ngẫu nhiên, và toàn bộ kênh dạng sóng được thay thế bằng bộ tạo số ngẫu nhiên nhị phân. Không cần điều khiển phần mô phỏng này với các bit thoại mã hoá.

Hiệu năng mã hoá kiểm soát lỗi cũng có thể được ước lượng dùng giải pháp bán giải tích, nó ánh xạ xác suất lỗi không mã hoá q thành xác suất lỗi mã hoá P_E. Kỹ thuật này đã được đề cập cho cả mã khối và mã xoắn ở chương 8. Theo đó, có thể ánh xạ phân bố q là hàm của E_b/N_0 thành phân bố của P_E là hàm của E_b/N_0.

Ước lượng số đo chất lượng thoại

Bước cuối cùng trong quá trình ước tính xác suất ngừng hoạt động là ước tính các phân bố số đo chất lượng thoại V đối với các giá trị khác nhau của E_b/N_0. Số đo chất lượng thoại sẽ

phụ thuộc vào P_E (chính nó phụ thuộc vào E_b/N_0) và phân bố của V là hàm của P_E. Vì vậy, ta có thể tìm được phân bố của V là hàm của E_b/N_0 bằng cách ước lượng số đo chất lượng thoại đối với các giá trị P_E khác nhau. Ước lượng này có thể được thực hiện độc lập với 2 phần đầu của bài toán: tất cả những gì ta phải làm là ước lượng hiệu năng của bộ mã hoá tiếng và bộ lập mã với các giá trị khác nhau của xác suất lỗi P_E. Tốt nhất là ta dùng tập chip bộ mã hoá/giải mã thoại thực tế, đưa thoại số hoá qua chúng và ước lượng chất lượng thoại đầu ra bộ giải mã là hàm của P_E. Ảnh hưởng của toàn bộ hệ thống giữa điểm B và G được *phỏng tạo* bằng cách tiêm các lỗi ngẫu nhiên tại tỉ lệ của P_E giữa đầu ra bộ mã hoá thoại và đầu vào bộ giải mã thoại. Phần ước lượng số đo chất chất lượng thoại này phải được thực hiện chỉ với hàng tá các giá trị P_E từ 10^{-1} đến 10^{-7} và người nghe phải cho điểm chất lượng thoại với mỗi tá giá trị P_E này, nó đơn giản hơn nhiều so với việc cho điểm chất lượng thoại đối với hàng ngàn điều kiện kênh trong mô phỏng Monte Carlo trực tiếp của toàn bộ hệ thống.



Hình 11.5: Phân chia hóa phương pháp luận để ước tính xác suất ngừng hoạt động

Cho trước ước tính số đo chất lượng thoại V là hàm của P_E có được từ phần III, phân bố P_E là hàm của q trong phần II và III, phân bố q là hàm của E_b/N_0 có được trong phần I, ta có thể nhận được V là hàm của E_b/N_0. Từ:

$$V\left(E_b/N_0\right)=V\left(P_e\left(q\left(E_b/N_0\right)\right)\right) \tag{11.5}$$

Ta có thể ước tính *phân bố* của V và xác suất ngừng hoạt động đối với mỗi giá trị của E_b/N_0. Từ đồ thị xác suất ngừng hoạt động theo E_b/N_0 ta có thể xác định giá trị tối thiểu của E_b/N_0 để đảm bảo xác suất ngừng hoạt động nhỏ hơn 2% tại chất lượng thoại nhỏ hơn 3. Giải pháp tổng thể để ước tính xác suất ngừng hoạt động được tóm tắt trong hình 11.5.

Tại đây, ta nghiên cứu chi tiết 3 phần ước tính lượng năng, bắt đầu bằng ước tính xác suất lỗi cho kênh dạng sóng. Đây là phần tập trung tính toán nhất, vì nó phải được lặp lại đối với 10 giá trị của E_b/N_0 và 10000 điều kiện kênh. Hai phần khác liên quan đến ánh xạ xác suất lỗi q đối với kênh dạng sóng thành xác suất lỗi mã hoá P_E và số đo chất lượng thoại V chỉ được lặp lại một lần đối với gần 10 giá trị q (10 giá trị E_b/N_0).

11.3.1. Phương pháp luận để mô phỏng phần tương tự của hệ thống

Mô hình mô phỏng kênh dạng sóng (phần tương tự của hệ thống) được được cho ở hình 11.6. Mục đích chính của mô phỏng là nhằm có được phân bố (hoành đồ) của xác suất lỗi q cho 10 giá trị khác nhau của E_b/N_0. Tại mỗi giá trị E_b/N_0, ta mô phỏng 10000 điều kiện chớp ảnh của kênh và nhận được hoành đồ của q từ ước tính BER đối với mỗi kênh. Cần phải đặc biệt lưu ý rằng, trong mỗi lần mô phỏng thì điều kiện kênh không thay đổi.

Ta mô tả các đặc điểm quan trọng nhất của mô hình mô phỏng trước khi trình bày các kỹ thuật tính toán hiệu quả để mô phỏng phần này của hệ thống.

Chi tiết hóa mô hình mô phỏng

Đầu vào: Đầu vào hệ thống gồm 2 nguồn nhị phân ngẫu nhiên, mỗi nguồn có tốc độ bit là 14400 bit/s (tốc độ kết hợp là 28800 bit/s), nó thể hiện luồng bit đến từ bộ mã hoá kiểm soát lỗi. Kết hợp hai chuỗi bit a_k và b_k thành chuỗi ký hiệu QPSK phức $\tilde{S}_k = A_k + jB_k$, trong đó A_k và B_k là sắp xếp các chuỗi nhị phân a_k và b_k thành các chuỗi biên độ +1 hoặc -1. Chuỗi ký hiệu QPSK được sắp xếp thành dạng sóng QPSK giá trị phức.

$$\tilde{x}(t)=\sum_{k=-\infty}^{\infty} A_k p(t-kT)+j\sum_{k=-\infty}^{\infty} B_k p(t-kT) \tag{11.6}$$

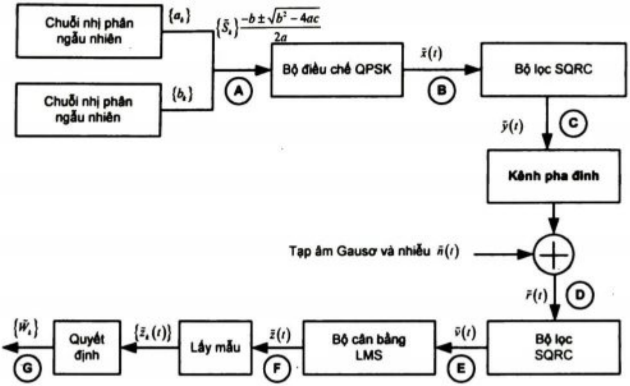
Được lấy mẫu tại tốc độ 12 mẫu/ký hiệu để tạo phiên bản mẫu của dạng sóng QPSK:

$$\tilde{x}(mT_s)=\sum_{k=-\infty}^{\infty} A_k p(mT_s-kT)+j\sum_{k=-\infty}^{\infty} B_k p(mT_s-kT) \tag{11.7}$$

Trong đó T là khoảng thời gian của ký hiệu QPSK, *p(t)* là xung chữ nhật biên độ đơn vị được định nghĩa là:

$$p(t)=\begin{cases}1, & 0 < t \le T\\ 0, & \text{nếu khác}\end{cases} \tag{11.8}$$

(kết hợp hai nguồn nhị phân ngẫu nhiên trong hình 11.6 thành một nguồn nhị phân ngẫu nhiên làm việc tại tốc độ kết hợp và nhận được 2 chuỗi riêng biệt bằng cách lấy các bit lẻ và chẵn tương ứng).



Hình 11.6: Mô hình mô phỏng cho phần tương tự của hệ thống

Các bộ lọc phát thu: Bộ lọc phát và bộ lọc thu là bộ lọc SQRC có hàm truyền đạt là:

$$H_T(f) = H_R(f) = \begin{cases} \sqrt{T}, & |f| \le \dfrac{1-\beta}{2T} \\ \sqrt{\dfrac{T}{2}\left[1-\sin\dfrac{\pi T}{\beta}\left(f-\dfrac{1}{2T}\right)\right]}, & \dfrac{1-\beta}{2T} \le |f| \le \dfrac{1+\beta}{2T} \\ 0, & |f| \ge \dfrac{1+\beta}{2T} \end{cases} \qquad (11.9)$$

Trong đó $\beta = 0{,}5$ đối với đường uốn 50% (*roll-off*). Các bộ lọc này là tối ưu theo nghĩa chúng tạo dạng sóng độ rộng băng thông hữu hạn với ISI = 0 và cũng cho ta hiệu năng BER tối ưu trên kênh AWGN. Trong hàm truyền đạt của bộ lọc phát thường chứa hàm *1/sinc* để bù dạng sóng QPSK (*là dạng xung NRZ chữ nhật chứ không phải là dạng xung kim*). Hàm truyền đạt bộ lọc được cho ở phương trình trên sẽ tạo ra đáp ứng có ISI = 0 chỉ khi đầu vào là một chuỗi các xung kim. Thay vì chứa hàm *1/sinc*, ta có thể dùng biểu diễn chuỗi xung kim của dạng sóng QPSK; trường hợp này, chỉ có mẫu đầu tiên trong 12 mẫu của ký hiệu QPSK thứ k có giá trị là A_k+jB_k và 11 mẫu còn lại là bằng 0.

Thực thi bộ lọc SQRC theo FIR bởi lẽ nếu thực hiện theo IIR rất khó (khó là vì hàm truyền đạt không được cho ở dạng điểm *không-cực* trong miền Laplace s như đã được đề cập ở chương 5). Ta sẽ coi rằng, dùng biến đổi bất biến xung kim bằng tích chập miền thời gian cho mỗi bộ lọc. Đáp ứng xung kim của bộ lọc SQRC được cho bởi:

$$h_T(t) = h_R(t) = 8\beta \frac{\cos\left[(R+2\beta)\pi t\right] + \sin\left[(R-2\beta)\pi t\right](8\beta t)^{-1}}{\pi\sqrt{R}\left[1-(8\beta t)^2\right]}, \quad -\infty < t < \infty \qquad (11.10)$$

Thấy rõ, đây là bộ lọc *không nhân quả*, nên để có được tính *nhân quả* ta cắt xét đáp ứng xung kim sau đó dịch thời gian tương ứng.

Kênh: Mô hình kênh *tựa-tĩnh (quasi-static)* được định nghĩa bởi (11.2), được đặc trưng bởi 2 biến ngẫu nhiên $\tilde{a}$ và τ. Mỗi mô phỏng được thực hiện với giá trị cố định của $\tilde{a}$ và τ (được rút ra từ phân bố Rayleigh và phân bố đều tương ứng). Giá trị của τ được làm tròn là một số nguyên lần các mẫu (*r* mẫu), vì thế mô hình mô phỏng kênh gồm một đường trực tiếp LOS và một đường trễ có độ trễ *r* mẫu tương ứng suy hao $\tilde{a}$. Đây là mô hình đơn giản.

Bộ cân bằng: Các bộ lọc SQRC sẽ tạo ra ISI = 0 chỉ khi hàm truyền đạt kênh là lý tưởng trên độ rộng băng thông của tín hiệu (trong ví dụ này nó bằng 0,75 lần tốc độ ký hiệu). Vì kênh trong trường hợp này là không lý tưởng, nên tồn tại ISI ≠ 0 trong hệ thống và có thể giảm thiểu ISI bằng bộ cân bằng trong máy thu. Tồn tại rất nhiều bộ cân bằng khả dụng, ta chọn bộ cân bằng chứa 9 nhánh, cụ thể là bộ cân bằng lỗi bình phương trung bình cực tiểu tuyến tính LMS (*Linear Minimum Mean Squared Error*), 9 nhánh cách đều đồng bộ để minh hoạ một số khía cạnh của phương pháp luận.

Thuật toán gradient thường được dùng để điều chỉnh các trọng số bộ cân bằng một cách lặp. Nếu mô phỏng tính hội tụ bộ cân bằng, thì phải thực thi theo phương pháp Monte Carlo dùng chuỗi hoa tiêu cho đầu vào và các mẫu tạp âm được thâm nhập vào trong quá trình mô phỏng. Vì bộ cân bằng LMS là bộ lọc tuyến tính và tạp âm tại đầu vào máy thu là AWGN, nên tạp âm tại đầu *ra* bộ cân bằng sẽ là tạp âm cộng và Gausơ, theo đó ta có thể dùng kỹ thuật bán phân tích để ước tính xác suất lỗi. Với ước tính BER, chỉ cần mô phỏng những ảnh hưởng của méo ISI còn những ảnh hưởng của tạp âm Gausơ trắng cộng có thể được xử lý theo cách giải tích mà không cần mô phỏng Monte Carlo các mẫu tạp âm.

Ta xét 2 giải pháp xử lý bộ cân bằng khi dùng kỹ thuật bán phân tích để ước tính BER:

Giải pháp 1, chạy mô phỏng thời gian ngắn lúc bắt đầu có chứa các mẫu tạp âm, đợi đến khi các trọng số bộ cân bằng hội tụ, sau đó "*ổn định lại*" các trọng số bộ cân bằng và thực hiện mô phỏng ước tính hiệu năng với nguồn tạp âm được tắt.

Giải pháp 2, dựa trên các trọng số bộ cân bằng sẽ hội tụ về véc tơ trọng số, các giá trị của véc tơ được tính toán giải tích theo:

$$\tilde{W} = \Gamma^{-1}R^{*} \qquad (11.11)$$

Trong đó $\tilde{W}$ là vectơ trọng số, Γ là "*ma trận hiệp biến kênh*" và R là vectơ các giá trị mẫu của đáp ứng xung kim không được cân bằng của hệ thống (từ đầu vào bộ lọc phát tới đầu ra bộ lọc thu). Đáp ứng xung kim không cân bằng này được lấy mẫu tại tốc độ ký hiệu, (có được từ việc *chạy hiệu chuẩn* trong đó một xung kim đơn vị được đưa vào bộ lọc phát và ghi lại đáp ứng xung kim tại đầu ra bộ lọc thu). Các giá trị mẫu của đáp ứng xung kim được dùng để tính toán hàm tự tương quan của đáp ứng xung kim không được cân bằng và các phần tử trong

ma trận Γ đạt được từ các giá trị của hàm tự tương quan. Các phần tử đường chéo trong Γ sẽ gồm giá trị tự tương quan tại trễ 0 cộng với phương sai tạp âm đầu vào bộ cân bằng (tính được các phần tử này nếu biết mật độ phổ công suất PSD tạp âm đầu vào và băng thông tạp âm của bộ lọc thu). Với giải pháp này, ta có thể tính toán các trọng số bộ cân bằng trước khi mô phỏng để ước tính BER như là một phần của "*quá trình hiệu chuẩn*" và bộ cân bằng được khảo sát như bộ lọc FIR trong quá trình mô phỏng BER.

Khi trực tiếp dùng mô phỏng Monte Carlo để ước tính hiệu năng, nguồn tạp âm sẽ được "*bật*" theo đó phương pháp lặp (gradient) được dùng tại lúc bắt đầu để các trọng số bộ cân bằng hội tụ. Sau đó, các trọng số này được cố định trong quá trình ước tính hiệu năng (nếu dùng phương pháp bán phân tích để ước tính BER thì tắt nguồn tạp âm trong giai đoạn ước tính BER bán phân tích).

Giải pháp Monte Carlo thuần tuý để ước tính hiệu năng

Trong giải pháp Monte Carlo trực tiếp, đầu vào và các quá trình tạp âm được mô phỏng một cách rõ ràng. Ước tính tỉ số lỗi tại mỗi giá trị của E_b/N_0 và điều kiện kênh bằng cách so sánh chuỗi ký hiệu đầu vào bộ điều chế $\tilde{S}_k$ và đầu ra thiết bị quyết định $\tilde{W}_k$ để xác định lỗi sau đó đếm lỗi. Trong khi bộ cân bằng tạo ra chuẩn hoá **biên độ** (không cần thiết đối với điều chế QPSK) và bù lệch pha, thì việc chạy hiệu chuẩn phải được thực hiện tại lúc bắt đầu để thiết lập chuẩn định thời cho bộ cân bằng và sắp các chuỗi ký hiệu đầu vào/ra thành hàng. Ngoài ra, phải dùng chuỗi hoa tiêu khởi đầu để hỗ trợ sự hội tụ bộ cân bằng, và ước tính tỉ số lỗi chỉ bắt đầu sau khi các trọng số bộ cân bằng hội tụ và giữ cố định. Các bước cơ bản trong mô phỏng Monte Carlo như sau:

1. Rút ra một tập $\tilde{a}$ và τ và bắt đầu với giá trị khởi đầu của E_b/N_0.

2. Chạy hiệu chuẩn để thiết lập chuẩn định thời cho bộ cân bằng và sắp hàng đầu vào/ra để đếm lỗi.

3. Duyệt bộ cân bằng và cô đọng lại (cố định) các trọng số (bật nguồn tạp âm với giá trị phương sai được tính từ E_b/N_0).

4. Bắt đầu mô phỏng Monte Carlo để ước tính hiệu năng và chạy mô phỏng cho đến khi đếm được khoảng 50 lỗi.

5. Lặp lại cho tất cả các giá trị của E_b/N_0 và 10.000 điều kiện kênh.

6. Tính toán hoành đồ của q cho mỗi giá trị của E_b/N_0.

Giải pháp Monte Carlo trực tiếp là giải pháp đơn giản để thực thi nguyên lý nhưng thời gian mô phỏng lâu tại mỗi giá trị của E_b/N_0 và điều kiện kênh. Thậm chí mỗi mô phỏng chỉ mất vài giây, thì lặp lại mô phỏng cho 10000 điều kiện kênh và 10 giá trị của E_b/N_0 là một vấn đề.

Vì tính tuyến tính của máy thu (bộ cân bằng LMS là bộ lọc FIR) và tạp âm là Gausơ & cộng tại đầu vào máy thu, nên tạp âm đầu ra cũng sẽ là tạp âm Gausơ cộng. Vì vậy, ta có thể dùng giải pháp bán phân tích để ước tính hiệu năng.

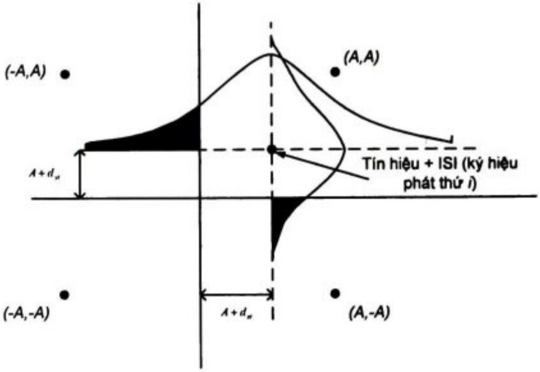
Giải pháp bán phân tích để ước tính hiệu năng

BER sẽ là *hàm của ISI và tạp âm Gausơ cộng* (các ảnh hưởng của chúng có thể được xử lý theo phép giải tích). Vì thế, chỉ cần mô phỏng ISI được tạo ra bởi sự nối tiếp của bộ lọc phát, kênh, bộ lọc thu và bộ cân bằng. Dùng giải pháp bán phân tích để ước tính BER (giả sử điểm

chòm sao tín hiệu phát là (1,1) được ánh xạ thành điểm (A,A) tại đầu ra bộ cân bằng như được thấy ở hình 11.7), cho ta:

$$\hat{q} = \frac{1}{M}\sum_{i=1}^{M}\left\{1-\left\{\left[1-Q\left(A+{d_{x_i}}/{\sigma_x}\right)\right]\times\left[1-Q\left(A+{d_{y_i}}/{\sigma_y}\right)\right]\right\}\right\}$$
$$\approx Q\left(A+{d_{x_i}}/{\sigma_x}\right)+Q\left(A+{d_{y_i}}/{\sigma_y}\right) \tag{11.12}$$

Trong đó d_{xi} và d_{yi} là thành đồng pha và vuông pha của ISI ứng với ký hiệu được mô phỏng thứ i, σ_x^2 và σ_y^2 là phương sai của thành phần đồng pha và vuông pha của tạp âm tại đầu ra bộ cân bằng, và M là số ký hiệu được mô phỏng.



Hình 11.7: Ước tính BER bán phân tích cho tín hiệu QPSK

Các giá trị của σ_x^2 và σ_y^2 được tính toán bởi:

$$\sigma_x^2 = \sigma_y^2 = N_0 B_N \tag{11.13}$$

Trong đó $N_0/2$ là PSD của tạp âm thông dải hai phía tại đầu vào bộ lọc thu, B_N là độ rộng băng thông tạp âm của bộ lọc thu và bộ cân bằng. Độ rộng băng thông tạp âm được tính từ việc chạy hiệu chuẩn.

Các bước áp dụng kỹ thuật bán phân tích để ước tính hiệu năng như sau:

1. Khởi đầu: Chọn một giá trị khởi đầu cho E_b/N_0 và các tham số kênh.

2. Hiệu chuẩn và xác định trọng số bộ cân bằng:

- Thiết lập chuẩn định thời cho bộ cân bằng và thời gian trễ tổng.

- Đạt được đáp ứng xung kim không cân bằng thông qua mô phỏng bằng cách đưa xung kim vào đầu vào A và sử dụng phương trình (11.11) để tính vectơ trọng số bộ cân bằng.

- Tính độ rộng băng thông tạp âm của bộ lọc thu, bộ cân bằng và dùng (11.13) để hiệu chuẩn phương sai tạp âm đầu ra.

3. Mô phỏng: Mô phỏng M ký hiệu và ước tính BER theo (11.12).

4. Lặp lại cho 10.000 kênh và 10 giá trị của E_b/N_0 và tính hoành đồ của q.

Kỹ thuật bán phân tích nhanh hơn

Ước tính tỉ số lỗi bán phân tích cho phép tăng đáng kể tốc độ bằng cách kết hợp tất cả các khối [bộ lọc phát, kênh, bộ lọc thu và bộ cân bằng (sau khi các trọng số đã được tính toán và thiết lập)] thành một khối duy nhất bởi lẽ chúng đều là các phần tử tuyến tính bất biến LTIV. Vì không có mẫu tạp âm nào được đưa vào, nên các khối này xử lý sóng QPSK theo kiểu đường ống. Với mục đích ước tính hiệu năng, chỉ cần quan tâm dạng sóng đầu ra bộ cân bằng. Trong trường hợp này, không cần xử lý tín hiệu vào/ra tại mỗi khối chức năng của hệ thống. Bằng cách kết hợp thành một khối và xử lý dạng sóng đầu vào, biểu diễn tương đương dẫn đến tính toán rất hiệu quả.

Đáp ứng xung kim toàn bộ hệ thống là:

$$\underbrace{\tilde{h}(kT_s)}_{\text{Hệ thống}} = \underbrace{\tilde{h}_T(kT_s)}_{\text{Bộ lọc phát}} \otimes \underbrace{\tilde{h}_c(kT_s)}_{\text{Kênh}} \otimes \underbrace{\tilde{h}_R(kT_s)}_{\text{Bộ lọc thu}} \otimes \underbrace{\tilde{h}_{eq}(kT_s)}_{\text{Bộ cân bằng}} \tag{11.14}$$

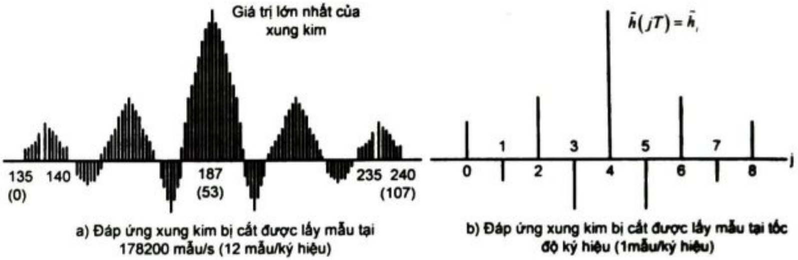
Đáp ứng này có thể đạt được một cách chính xác thông qua mô phỏng bằng cách đưa *xung kim* vào điểm A và đo đáp ứng xung kim tại đầu ra của bộ cân bằng (điểm F trong hình 11.4). Đáp ứng xung kim toàn bộ có thể được cắt bớt và toàn bộ hệ thống có thể được mô phỏng như *một* bộ lọc FIR. Chẳng hạn, đáp ứng xung kim toàn bộ được cho ở hình 11.8. Lưu ý rằng, trễ qua hệ thống xấp xỉ bằng 135 mẫu và đáp ứng xung kim có thể bị cắt bớt còn 108 mẫu (từ mẫu thứ 135 đến mẫu thứ 242). Đáp ứng xung kim nhận giá trị 0 ở bên ngoài khoảng này (xem hình 11.8). Chỉ số thời gian cho các giá trị khác 0 của đáp ứng xung kim được đánh số lại từ 0 đến 107 để tiện ký hiệu.

Mô phỏng mô hình mức dạng sóng chi tiết sẽ được thực hiện tại tốc độ lấy mẫu $r_S = 172.800$ mẫu/s theo phương trình:

$$\tilde{z}(mT_s) = \sum_{p=0}^{107} \tilde{h}(pT_s).\tilde{x}((m-p)T_s) \tag{11.15}$$

Trong đó $\tilde{x}(mT_s)$ là các giá trị mẫu của dạng sóng QPSK tại đầu vào bộ lọc phát, $\tilde{z}(mT_s)$ là đầu ra bộ cân bằng, $\tilde{h}(pT_s)$ là các giá trị của đáp ứng xung kim chung bị xén bớt. Đầu ra của bộ cân bằng được lấy mẫu bắt đầu tại mẫu số 187 (xem hình) và mỗi khi 12 mẫu trôi qua tạo ra số đo quyết định $\tilde{z}_k$ và dựa trên giá trị của số đo quyết định để thực hiện một quyết định $\tilde{W}_k$ (ước tính của ký hiệu phát). Với ước tính hiệu năng, ta chỉ quan tâm sự xuất hiện mẫu thứ 12 (do lấy mẫu là 12 mẫu/ký hiệu) tại đầu ra bộ cân bằng, tương ứng với các thời điểm quyết định (các mẫu ở giữa không dùng). Vì bộ cân bằng hoạt động với khoảng cách nhánh trễ là 12 mẫu (khoảng thời gian một ký hiệu), nên ta có thể viết biểu thức cho số đo quyết định dùng mẫu thứ 12 của đáp ứng xung kim (xem hình 11.8) là:

$$\tilde{z}_i = \underbrace{\sum_{j=0}^{8} \tilde{h}_j \tilde{S}_{i-j}}_{\text{Lưu ý: Lấy mẫu đáp ứng xung kim \& dịch tín hiệu}} \tag{11.16}$$



Hình 11.8: Đáp ứng xung kim bị cắt bớt được lấy mẫu tại 12 mẫu trên ký hiệu và 1 mẫu trên ký hiệu

Lưu ý rằng, (11.16) là mô hình mô phỏng toàn bộ để ước tính tỉ số lỗi bán phân tích; ta chỉ cần tạo một chuỗi ký hiệu QPSK và xử lý chúng theo (11.16). Dẫn đến chỉ cần 8 phép toán trên một ký hiệu để tạo các giá trị của số đo quyết định, nó thể hiện ký hiệu đầu vào với ISI cộng. Tính toán lỗi bán phân tích theo (11.12) được áp dụng cho chuỗi $\tilde{z}_i$.

Thấy rõ, mô hình theo (11.16) sẽ nhanh hơn mô hình theo (11.15), vì ở mô hình (11.16) việc tính đầu ra chỉ cần mẫu thứ 12 và với mỗi mẫu đầu ra chỉ cần 8 phép nhân và 8 phép cộng *so với* 108 phép nhân và 108 phép cộng đối với mô hình (11.15). So sánh xử lý dạng sóng QPSK qua mỗi khối: mô hình kết hợp theo (11.15) sẽ nhanh hơn 3 đến 4 lần (vì ta kết hợp 4 khối thành một). Vì vậy, mô hình mô phỏng theo (11.16) nhanh hơn khoảng 1000 lần.

Mô hình mô phỏng theo (11.16), kết hợp với giải pháp bán phân tích để ước tính BER mang lại hiệu quả tính toán. Chiều dài bộ nhớ chung của hệ thống là 9 ký hiệu dẫn đến chu kỳ chuỗi PN là $2^9 = 512$ ký hiệu sẽ đủ để tạo tất cả các giá trị ISI có thể có, tất nhiên loại trừ chuỗi toàn 0. Vì vậy, sau khi hiệu chuẩn, mỗi mô phỏng hiệu năng bao gồm việc tạo 512 ký hiệu QPSK, tính 512 mẫu đầu ra theo (11.16) và ước tính bán phân tích theo (11.16).

Tóm tắt các bước ước tính tỉ số lỗi phân tích nhanh như sau:

1. Khởi tạo: Chọn một giá trị khởi tạo cho E_b/N_0 và các tham số kênh.

2. Hiệu chuẩn và xác định trọng số bộ cân bằng:

- Thiết lập chuẩn định thời cho bộ cân bằng và độ trễ thời gian tổng.

- Đạt được đáp ứng xung kim không cân bằng thông qua mô phỏng bằng cách đưa xung kim vào đầu vào A và dùng (11.11) để tính toán vectơ trọng số bộ cân bằng.

- Tính toán băng thông tạp âm bộ lọc thu, bộ cân bằng và dùng (11.13) để hiệu chuẩn phương sai tạp âm tại đầu ra.

- Đạt được đáp ứng xung được cân bằng được lấy mẫu tại tốc độ ký hiệu bằng cách đưa xung kim vào điểm A và lấy mẫu đáp ứng xung kim đầu ra bộ cân bằng 1mẫu/1ký hiệu (xem hình 11.8).

3. Mô phỏng: Tạo M = 512 ký hiệu QPSK, xử lý chúng theo (11.16) và ươc tính xác suất lỗi theo (11.12).

4. Lặp lại cho 10000 kênh và 10 giá trị E_b/N_0, sau đó tính hoành đồ của q.

Phương pháp momen để ước tính BER

Kỹ thuật bán phân tích nhanh cho ta hiệu quả tính toán nhưng khối lượng tính toán tăng đáng kể nếu chiều dài nhớ của hệ thống tăng. Việc tăng chiều dài nhớ (vì số các ký hiệu cần được mô phỏng cho tất cả các giá trị ISI) tăng theo $M = m^L$, trong đó L là chiều dài nhớ của hệ thống và m là kích thước chữ cái (trường hợp nhị phân $m = 2$). Khi m và L lớn thì thời gian mô phỏng sẽ rất lâu. Những trường hợp như vậy có thể dùng phương pháp khác để giảm tải tính toán.

Tính toán cơ bản được thực hiện trong (11.12) được biểu diễn:

$$\hat{q} = E\left\{Q\left(A + \frac{D_x}{\sigma_x}\right)\right\} + E\left\{Q\left(A + \frac{D_y}{\sigma_y}\right)\right\} \tag{11.17}$$

Trong đó lấy kỳ vọng toán học theo D_x và D_y, chúng là các biến ngẫu nhiên thể hiện thành phần đồng pha và vuông pha của ISI. Không phải là ước tính giá trị kỳ vọng này bằng cách lấy trung bình các giá trị mô phỏng của ISI mà ta tính toán các momen của ISI, lấy xấp xỉ phân bố của D_x và D_y dùng các momen của ISI và sau đó thực hiện phép kỳ vọng dùng phân bố xấp xỉ của ISI. Muốn vậy

Lấy xấp xỉ phân bố của D_x dùng các momen của nó. Từ (11.16), có thể viết ISI theo D_x là:

$$D_x = \sum_{\substack{j=0 \\ j\neq 0}}^{8} \alpha_j A_j - \sum_{j=0}^{8} \beta_j B_j \tag{11.18}$$

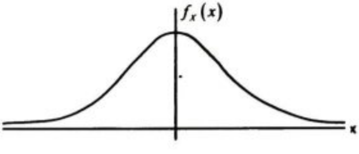
Trong đó α_j và β_j là phần thực và phần ảo của đáp ứng xung kim (nghĩa là $\tilde{h}_k = \alpha_k + j\beta_k$), $A_j = \pm 1$, $B_j = \pm 1$ là phần thực và phần ảo của chuỗi ký hiệu QPSK. Các momen của D_x có thể được tính theo:

$$E\left\{D_x^k\right\} = E\left\{\left[\sum_{\substack{j=0 \\ j\neq 0}}^{8} \alpha_j A_j - \sum_{j=0}^{8} \beta_j B_j\right]^k\right\}, \qquad k = 1, 2, 3... \tag{11.19}$$

Lưu ý rằng, trong phương trình trên, các giá trị của đáp ứng xung kim là những hằng số các giá trị của nó được biết, A_j và B_j là các biến ngẫu nhiên nhị phân độc lập có giá trị ± 1. Các momen lẻ của A_j & B_j là bằng 0 *và* các momen chẵn của A_j & B_j là bằng 1. Theo đó, tính toán các momen của D_x gồm khai triển tổng nhị thức trong (11.19). Ta lấy kỳ vọng cho từng số hạng và cộng chúng lại.

Từ các momen của D_x, ta được xấp xỉ rời rạc cho phân bố của D_x. Trong xấp xỉ này, D_x được xử lý như biến ngẫu nhiên rời rạc có J giá trị $d_1,..., d_J$, với các xác suất $p_1,...,p_J$. Như một ví dụ minh họa đơn giản của nguyên lý, xấp xỉ hóa rời rạc của hàm mật độ xác suất pdf phân bố Gausơ được cho ở hình 11.9.

Chọn $x_1,..x_J$ và $p_1,...p_J$ sao cho phân bố liên tục và phân bố xấp xỉ rời rạc tương ứng tạo ra các momen như nhau.



Hình 11.9: Ví dụ xấp xỉ rời rạc của pdf Gausơ

$$E\left(X^n\right) = \mu_n = \sum_{k=1}^{J} x_k^n p_k \tag{11.20}$$

Cho trước $2J$ momen đầu tiên $\mu_1, \ldots, \mu_{2J}$ của X, ta có thể giải tập $2J$ phương trình phi tuyến:

$$\mu_n = \sum_{k=1}^{J} x_k^n p_k \quad n = 1, 2 \ldots 2J \tag{11.21}$$

Đối với các giá trị của x_k và p_k, $k = 1,2,\ldots J$. Chi tiết về các kỹ thuật được dùng để rút ra xấp xỉ rời rạc của phân bố theo các momen của nó được thấy ở nhiều tài liệu.

Dùng xấp xỉ rời rạc đối với phân bố ISI ta có thể tính $E\left\{Q\left(A + D_x / \sigma_x\right)\right\}$ là:

$$E\left\{Q\left(A + D_x / \sigma_x\right)\right\} \approx \sum_{j=1}^{J} Q\left(A + d_j / \sigma_x\right) p_j \tag{11.22}$$

Số hạng thứ hai trong (11.17) có thể được tính dùng thủ tục tương tự.

Lưu ý rằng, trong phương pháp này, không dùng mô phỏng Monte Carlo cho tất cả ước tính hiệu năng!. Nó hoàn toàn là giải tích *ngoại trừ* 2 *mô phỏng sự kiện đơn để có được*: ***(i)*** đáp ứng xung không cân bằng từ việc tính toán các trọng số bộ cân bằng; ***(ii)*** đáp ứng xung cân bằng được lấy mẫu tại tốc độ ký hiệu. Dùng (11.19) để tính toán các momen của phân bố ISI và xấp xỉ rời rạc của phân bố của mô phỏng ISI và xác suất lỗi cũng được *tính toán*. Ngoài 2 mô phỏng sự kiện đơn, ta phải chạy chương trình hiệu chuẩn để thiết lập các chuẩn định thời nhằm lấy mẫu các xung không cân bằng, cân bằng và tính toán băng thông tạp âm của bộ lọc máy thu.

Hiệu quả tính toán của phương pháp momen phụ thuộc vào số lượng momen cần đến để đạt được xấp xỉ phân bố ISI chính xác và khối lượng tính toán các momen. Tính toán sau sẽ là một hàm của chiều dài đáp ứng xung kim (trong nhiều trường hợp có thể được cắt xén 10 hoặc nhiều ký hiệu). Các xấp xỉ tin cậy của phân bố ISI có thể đạt được từ 6 hoặc 8 momen đầu tiên.

Phương pháp momen rất hữu hiệu khi dùng các sơ đồ điều chế bậc cao (chẳng hạn QAM 256). Trong trường hợp này dạng sóng đồng pha và vuông pha có 16 mức biên độ, và nếu chiều dài nhớ là 10, thì ta cần chuỗi PN 16-ary dài 16^{10} ký hiệu để mô phỏng tất cả các giá trị ISI có thể có. Theo đó, phương pháp bán phân tích sẽ yêu cầu thời gian mô phỏng dài, trong khi phương pháp momen trong trường hợp này hiệu quả tính toán cao hơn nhiều.

Tóm tắt các bước then chốt về ứng dụng phương pháp momen như sau:

1. Khởi tạo: Chọn một giá trị khởi tạo cho E_b/N_0 và các tham số kênh.

2. Hiệu chuẩn và xác định trọng số bộ cân bằng:

- Thiết lập chuẩn định thời cho bộ cân bằng và thời gian trễ tổng.

- Đạt được đáp ứng xung kim không cân bằng thông qua mô phỏng bằng cách đưa xung kim vào đầu vào A và dùng (11.11) để tính toán vectơ trọng số cho bộ cân bằng.

- Tính toán băng thông tạp âm của bộ lọc thu và bộ cân bằng và dùng (11.13) để hiệu chuẩn phương sai của tạp âm tại đầu ra.

- Đạt được đáp ứng xung cân bằng được lấy mẫu tại tốc độ ký hiệu bằng cách đưa xung kim vào (điểm A trong hình 11.4) và lấy mẫu đáp ứng xung kim đầu ra bộ cân bằng một mẫu/mỗi ký hiệu (xem hình 11.8).

3. Tính toán BER:

- Tính các momen và monen dựa trên phép lấy xấp xỉ phân bố của ISI.

- Tính BER theo (11.22).

4. Lặp lại cho 10000 kênh và 10 giá trị của E_b/N_0 và tính hoành đồ của q.

11.3.2. Tóm tắt phương pháp luận để mô phỏng phần tương tự của hệ thống

Phần này ta đã minh hoạ các khía cạnh quan trọng của phương pháp luận để mô phỏng phần xử lý dạng sóng của hệ thống thông tin vô tuyến làm việc trong môi trường kênh *pha đinh chậm*. Đề cập một vài giải pháp để đơn giản hóa bài toán mô phỏng *và* cũng trình bày một số kỹ thuật ước tính hiệu năng. Các kỹ thuật này: từ Monte Carlo thuần tuý đến Monte Carlo từng phần, đến phương pháp phân tích hoàn toàn.

Trong mọi mô phỏng ước tính hiệu năng, cần phải tận dụng triệt để các nỗi lực trước để đơn giản mô hình mô phỏng, để có thể kiểm tra và đánh giá các giải pháp khác nhau. Những nỗ lực này sẽ tiết kiệm tính toán rất nhiều khi mô phỏng ước tính hiệu năng. Nhiều chi tiết được thảo luận ở dạng tình huống của ví dụ, tuy không ứng dụng trực tiếp vào các bài toán khác nhưng thể hiện tính toàn diện của phương pháp luận, gợi mở nhiều nhân tố phải xét đến trước quyết định các quy trình ước tính hiệu năng.

Tại đây cho thấy rõ: *Việc hiệu chuẩn và mô phỏng sự kiện đơn để đo đáp ứng xung và đáp ứng xung kim* có vài trò quan trọng trước khi thực hiện mọi ước tính hiệu năng. Mọi hiệu chuẩn đều không phải lặp lại cho mỗi tình huống. Chẳng hạn nếu điều kiện kênh là không đổi thì không phải đo đáp ứng xung không cân bằng với mỗi giá trị của E_b/N_0.

11.3.3. Ước tính BER được mã hoá

Bước tiếp theo trong quá trình ước tính xác suất ngừng hoạt động là để nhằm có được xác suất lỗi bit mã hoá $P\left(E_b/N_0\right)$ từ xác suất lỗi bit không mã hoá $q\left(E_b/N_0\right)$. Điều này có thể được thực hiện theo cách bán phân tích bằng cách dùng giới hạn hàm truyền đạt cho mã xoắn (được dùng để sửa lỗi mà đã được đề cập ở chương 8).

11.3.4. Ước tính số đo chất lượng thoại

Bước cuối cùng trong quá trình ước tính xác suất ngừng hoạt động là ánh xạ xác suất lỗi mã hoá thành số đo chất lượng thoại. Điều này được thực hiện bằng cách phun các lỗi nhị phân

giữa đầu ra bộ mã hoá thoại và đầu vào bộ giải mã hoá và đánh giá chất lượng đầu ra thoại kết quả. Bằng cách lặp lại các thử nghiệm nghe cho các giá trị khác nhau của tỉ lệ lỗi được phun vào, ta có thể thiết lập quan hệ giữa tỉ số lỗi P_E và số đo chất lượng thoại V. Một ví dụ minh họa được cho ở bảng 11.1.

Bảng 11.1: Quan hệ giữa xác suất lỗi và chất lượng thoại

P_E	Chất lượng thoại V
$> 10^{-1}$	1
10^{-1} đến 5.10^{-3}	2
5.10^{-3} đến 10^{-4}	3
10^{-4} đến 10^{-6}	4
$< 10^{-6}$	5

Lưu ý rằng đối với phần ước lượng hiệu năng, rút ra quan hệ giữa P_E và V được thực hiện độc lập với các bước trước đó. Vì vậy, trong bảng này có thể thấy từ nhà sản xuất chip mã hoá/giải mã thoại.

Từ bảng 11.1 thấy rõ, có thể biểu diễn xác suất ngừng hoạt động P(V < 3) theo phân bố của P_E là:

$$\Pr(V < 3) = \Pr\{P_e > 5\times 10^{-3}\} \tag{11.23}$$

Từ các giới hạn giải tích (nó quan hệ hóa giữa P_E và q), ta có thể thiết lập giá trị của q là q_o tạo ra $P_E > 5\times 10^{-3}$. Sau đó, xác suất ngừng hoạt động hệ thống là $P(q > q_o)$, tìm được các giá trị của E_b/N_0 từ phân bố của q BER đối với phần tương tự của kênh. Xác suất ngừng hoạt động $P(q > q_o)$ sẽ giảm khi E_b/N_0 tăng, và bằng cách vẽ xác suất ngừng hoạt động theo E_b/N_0 (tức là, xác suất ngừng hoạt động là hàm của E_b/N_0), ta có thể xác định được giá trị cực tiểu của E_b/N_0 cần thiết để duy trì xác suất ngừng hoạt động dưới 2% *(nghĩa là $P(q < q_o) < 0{,}02$).*

11.3.5. Tổng kết phương pháp luận toàn diện

Tại đây, ta tóm tắt các bước chính trong phương pháp luận để ước tính xác suất ngừng hoạt động trong hệ thống thông tin vô tuyến:

1. Xác định quan hệ giữa xác suất lỗi mã hoá P_E và số đo chất lượng thoại V bằng cách mô phỏng mã hoá/giải mã thoại với các giá trị khác nhau của P_E.

2. Tính toán các giới hạn để quan hệ hóa giữa xác suất lỗi không mã hoá q và xác suất lỗi mã hoá P_E.

3. Mô phỏng phần tương tự của hệ thống với các giá trị khác nhau của E_b/N_0 và 10.000 điều kiện kênh và đạt được phân bố q đối với các giá trị khác nhau của E_b/N_0.

4. Ánh xạ phân bố của q sang phân bố của V và với các giá trị khác nhau của E_b/N_0 và xác định giá trị của Eb/N0 cần thiết để duy trì xác suất ngừng hoạt động đã định.

Chương 12
MÔ HÌNH HÓA VÀ MÔ PHỎNG THIẾT BỊ PHI TUYẾN

12.1. Mở đầu

Dựa vào tính phụ thuộc vào thời gian và không phụ thuộc vào thời gian (bất biến), tính tuyến tính và phi tuyến, tính nhân quả và không nhân quả, và với mục đích nghiên cứu, phân loại mô hình mô phỏng khối chức năng trong hệ thống truyền thông thành 3 loại:

1. Hệ thống nhân quả tuyến tính bất biến LTIV như: bộ lọc và kênh truyền thông tĩnh

2. Hệ thống tuyến tính thay đổi theo thời gian LTV như: kênh pha đinh và bộ cân bằng tuyến tính

3. Hệ thống bất biến phi tuyến như: bộ khuếch đại công suất cao, bộ giới hạn và vòng khoá pha PLL.

Tuy hầu hết các khối chức năng trong hệ thống truyền thông là tuyến tính hoặc xấp xỉ tính cách tuyến tính nhưng tồn tại một số khối phi tuyến. Một số vốn đã phi tuyến và được dùng một cách chủ định để cải thiện hiệu năng hệ thống như: Bộ giới hạn là phần tử phi tuyến đặt trước máy thu để cải thiện hiệu năng khi xuất hiện tạp âm xung kim. Bộ cân bằng hồi tiếp quyết định DFE là thiết bị phi tuyến cho ta hiệu năng tốt hơn so với bộ cân bằng tuyến tính. Tuy nhiên, các khối chức năng khác như bộ khuếch đại công suất cao biểu lội tính phi tuyến không mong muốn. Nhiều khối chức năng trong hệ thống truyền thông được thiết kế để thực hiện các chức năng tuyến tính nhưng trong quá trình hoạt động cụ thể có thể gây ra các hiệu ứng phi tuyến.

Phân tích toán học các hiệu ứng phi tuyến thường rất khó, thậm chí trong trường hợp phi tuyến không nhớ bậc ba đơn giản như:

$$y(t) = x(t) - 0,2x^3(t) \tag{12.1}$$

Ở dạng tổng quát rất khó tính toán hàm mật độ xác suất pdf và hàm tự tương quan đầu ra $y(t)$ từ hàm pdf và hàm tự tương quan đầu vào $x(t)$, nhưng mô phỏng dễ dàng tạo các giá trị mẫu đầu vào $\{x(kTs)\}$ sau đó dùng (12.1) tạo đầu ra $\{y(kTs)\}$. Từ hai chuỗi $\{x(kTs)\}$ và $\{y(kTs)\}$ có thể ước tính một số đại lượng như: hàm mật độ xác suất pdf và hàm tự tương quan của $x(t)$, $y(t)$ cũng như hàm tương quan chéo giữa $x(t)$ và $y(t)$. Thực vậy, mô phỏng là phương pháp hữu hiệu để phân tích và thiết kế các hệ thống truyền thông chứa các phần tử phi tuyến như: các bộ lọc không lý tưởng, tạp âm phân bố phi Gausơ. Chương này tập trung các phương pháp lập mô hình và mô phỏng các phần tử phi tuyến trong các hệ thống truyền thông.

12.1.1. Các loại thiết bị phi tuyến và các mô hình

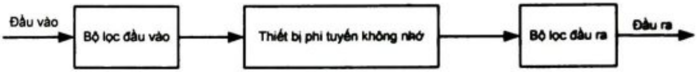
Dựa vào dải tần làm việc, ta phân loại và mô hình hóa thiết bị phi tuyến

Các thiết bị phi tuyến trong hệ thống truyền thông có thể hoặc là thiết bị băng tần cơ sở hoặc là thiết bị thông dải. Chẳng hạn: bộ giới hạn là một thiết bị phi tuyến băng tần cơ sở trong

khi đó bộ khuếch đại cao tần RF là một thiết bị phi tuyến thông dải. Nếu đưa một tín vào thiết bị phi tuyến thông dải được trung tâm tại tần số f_c thì đầu ra sẽ có phổ nằm ở lân cận tần số f_c. Các thành phần hài gần tần số $2f_c$, $3f_c$... hầu hết không được quan tâm vì các khối chức năng "*luồng xuống*" từ phần tử phi tuyến thường loại bỏ các thành phần này. Mô hình được dùng phổ biến nhất cho thiết bị phi tuyến thông dải là một thiết bị phi tuyến băng tần cơ sở theo sau là *"bộ lọc vùng "*, bộ lọc chỉ cho qua các thành phần nằm gần hoặc cạnh dải băng tần tín hiệu đầu vào.

Dựa vào tính có nhớ và không nhớ, ta phân loại và mô hình hóa thiết bị phi tuyến

Như một cách phân loại khác cho thiết bị phi tuyến là thiết bị đó có nhớ hay không có nhớ. Đầu ra của thiết bị phi tuyến không nhớ tại thời điểm t chỉ phụ thuộc vào giá trị tức thời của đầu vào tại thời điểm t, trong khi đó thiết bị phi tuyến có nhớ sẽ tạo tín hiệu đầu ra tại thời điểm t là một hàm của đầu vào tại thời điểm hiện tại và quá khứ. Thiết bị có nhớ sẽ bộc lộ *tính cách chọn lọc tần số* và thường được mô hình hóa bằng một thiết bị phi tuyến không nhớ được *"kẹp"* giữa hai bộ lọc như được cho ở hình 12.1. Bộ lọc giải thích cho tính chọn lọc tần số của thiết bị phi tuyến có nhớ.



Hình 12.1: Mô hình hóa cho thiết bị phi tuyến chọn lọc tần số có nhớ

Thiết bị phi tuyến thông dải có thể được mô hình hóa và mô phỏng bằng cách sử dụng phiên bản thông dải giá trị thực của tín hiệu. Sẽ được sáng tỏ, tính cách của hầu hết các phần tử phi tuyến thông dải đều có thể được mô hình hóa và mô phỏng dưới dạng các đường đường bao phức thông thấp của các tín hiệu thông dải. Việc biểu diễn đường bao phức thông thấp dẫn đến hiệu quả tính toán, do đó được dùng nhiều hơn.

Thiết bị phi tuyến cũng có thể được mô tả bởi phương trình vi phân phi tuyến. Khi này, mô phỏng có thể được thực hiện theo nghiệm đệ qui của phương trình vi phân phi tuyến. Một cách tiếp cận tương đương là trình bày hệ thống phi tuyến ở dạng sơ đồ khối (nếu có thể được), sau đó mô phỏng mô hình sơ đồ khối đó. Giải pháp này thường được gọi là phương pháp sơ đồ khối lắp ráp.

Phương pháp sơ đồ khối lắp ráp được trình bày chi tiết trong chương 6 khi xét vòng khoá pha, một cách ngắn gọn ta có thể mô tả một phân hệ bằng phương trình vi phân:

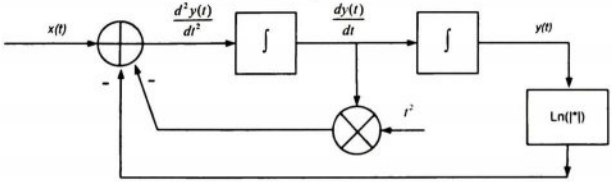
$$\frac{d^2y(t)}{dt^2}+t^2\frac{dy(t)}{dt}+\ln\left(|y(t)|\right)=x(t) \tag{12.2}$$

Được xắp xếp lại là:

$$\frac{d^2y(t)}{dt^2}=-t^2\frac{dy(t)}{dt}-\ln\left(|y(t)|\right)-x(t) \tag{12.3}$$

Được trình bày ở dạng sơ đồ khối như ở hình 12.2. Như đã rõ từ chương 6, mô phỏng tức thì theo cách trình bày tích phân liên tục theo thời gian được xấp xỉ bởi thuật toán tích phân rời rạc theo thời gian.

Phương trình vi phân phi tuyến (12.2) có thể được mô phỏng trực tiếp sử dụng kỹ thuật tích phân số đệ qui. Phương pháp này mang lại hiệu quả tính toán nhưng cần có một số nỗ lực về việc lập mô hình. Vì thế, nó thường được sử dụng cho các thiết bị phi tuyến rất phức tạp.



Hình 12.2: Mô hình sơ đồ khối lắp ráp cho thiết bị phi tuyến

12.1.2. Các nhân tố cần xét đến khi mô phỏng thiết bị phi tuyến

Cần phải xét một số nhân tố khi mô phỏng thiết bị phi tuyến. Mô phỏng thiết bị phi tuyến hầu hết được thực hiện trong miền thời gian ngoại trừ bộ lọc (bộ lọc được sử dụng trong mô hình để giải thích cho tính cách chọn lọc tần số). Tất nhiên, có thể mô phỏng bộ lọc trong miền thời gian và miền tần số.

Để minh họa một số nhân tố trong trình bày lập mô hình hoặc mô phỏng thiết bị phi tuyến, ta giả thiết rằng mô hình được xét là mô hình phi tuyến không nhớ có dạng được mô tả bởi (12.1) hoặc mô hình chọn lọc tần số (có nhớ) được cho ở hình 12.1.

Tốc độ lấy mẫu

Nhân tố đầu tiên cần phải xem xét là tốc độ lấy mẫu. Với hệ thống tuyến tính, ta thường thiết lập tốc độ lấy mẫu từ 8 đến 16 lần độ rộng băng tần tín hiệu vào. Trường hợp phi tuyến dạng:

$$y(t) = x(t) - 0,2x^3(t) \tag{12.4}$$

Trong đó đầu vào x(t) là tín hiệu năng lượng tất định, chuyển sang miền tần số:

$$Y(f) = X(f) - 0,2X(f) \otimes X(f) \otimes X(f) \tag{12.5}$$

Trong đó, ⊗ là phép tích chập. Tích chập bội ba dẫn đến độ rộng băng tần đầu ra *Y(f)* lớn hơn độ rộng băng tần đầu vào *X(f)*. Hiệu ứng này được gọi là trải phổ do tính phi tuyến gây ra. Nếu muốn trình bày *y(t)* một cách thích đáng mà không có lỗi chồng phổ quá mức, thì tốc độ lấy mẫu phải dựa trên độ rộng băng tần của *y(t)* (độ rộng băng tần của *y(t)* lớn hơn nhiều độ rộng băng tần của *x(t)*). Theo đó, khi thiết lập tốc độ lấy mẫu để mô phỏng thiết bị phi tuyến cần phải tính đến ảnh hưởng này. Tuy nhiên, tần số lấy mẫu thực tế cần thiết để mô phỏng không lớn như ví dụ này.

Nối tầng

Một nhân tố khác phải tính đến là ảnh hưởng của việc nối tầng các khối tuyến tính và phi tuyến như hình 12.1. Chẳng hạn, nếu đang dùng kỹ thuật chồng chéo (overlap) và cộng để mô phỏng bộ lọc, ta phải cẩn thận. Không thể xử lý các khối dữ liệu qua bộ lọc thứ nhất, sau đó là

qua thiết bị phi tuyến và bộ lọc thứ hai và thực hiện chồng chéo và cộng tại đầu ra bộ lọc thứ hai. Làm như vậy là sai bởi lẽ nguyên lý xếp chồng không ứng dụng được cho thiết bị phi tuyến. Xử lý đúng là áp dụng phương thức chồng chéo và cộng tại đầu ra của bộ lọc thứ nhất, tính toán các mẫu trong miền thời gian tại đầu ra bộ lọc thứ nhất, xử lý mẫu này trên cơ sở từng mẫu một qua thiết bị phi tuyến không nhớ và ứng dụng kĩ thuật chồng chéo và cộng cho bộ lọc thứ hai.

Vòng hồi tiếp phi tuyến

Các vòng hồi tiếp có thể cần thêm trễ một mẫu trong vòng để tránh hiện tượng treo trong tính toán (lưu ý đường hồi tiếp trong PLL ở chương 6). Trễ nhỏ trong một vòng hồi tiếp tuyến tính có thể không ảnh hưởng đến kết quả mô phỏng nhưng trong hệ thống phi tuyến, có thể không những làm suy thoái đáng kể kết mô phỏng mà còn dẫn đến hoạt động bất ổn định. Để tránh các ảnh hưởng này ta phải tăng tốc độ lấy mẫu (tương đương giảm độ trễ).

Tốc độ lấy mẫu khả biến và nội suy

Nếu mô hình là một phương trình vi phân phi tuyến được giải bằng kỹ thuật tích phân số, thì rất nhiều thuật toán tích phân số trong các gói phần mềm (như Simlink) sử dụng kích cỡ bước tích phân khả biến. Kích cỡ bước được xác định tự động tại mỗi bước thực hiện tùy thuộc vào tính cách của nghiệm. Nếu nghiệm có tính cục bộ tốt, thì có thể dùng kích cỡ bước lớn. Để tránh chồng phổ trong các khối luồng xuống, cần thực hiện nội suy đầu ra và tạo các mẫu tín hiệu đầu ra cách đều.

12.2. Mô hình hóa và mô phỏng thiết bị phi tuyến không nhớ

Quan hệ vào/ra của thiết bị phi tuyến không nhớ được cho bởi:

$$y(t) = F\big(x(t-t_0)\big) \tag{12.6}$$

Trong đó, *F(.)* là ***một hàm phi tuyến*** và t_0 có thể được giả thiết bằng 0. Nếu *x(t)* là tín thiệu băng tần cơ sở, thì *y(t)* cũng được xử lý là tín hiệu băng tần cơ sở. Mặt khác, nếu *x(t)* là một tín hiệu thông dải, thì *y(t)* sẽ là tín hiệu thông dải nhưng *y(t)* cũng có thể chứa thành phần một chiều và các thành phần hài của tín hiệu đầu vào. Trong các trường hợp cụ thể, ta chỉ quan tâm những thành phần đầu ra có phổ gần các tần số đầu vào, thực hiện bằng cách đặt một bộ lọc thông dải vùng tại đầu ra thiết bị phi tuyến. Thiết bị phi tuyến tự nó được mô hình như một thiết bị hoạt động tức thời đơn giản.

Trong chương 4 ta đã biểu diễn đường bao phức thông thấp để mô hình hóa và mô phỏng tín hiệu thông dải qua hệ thống tuyến tính. Tại đây, ta cũng muốn có mô hình cho thiết bị phi tuyến thông dải ở dạng đường bao thông thấp phức của đầu vào/ra. Ta rút ra mô hình đường bao thông thấp phức cho các thiết bị phi tuyến như: bộ hạn chế, bộ khuếch đại cao tần RF băng rộng, vòng khóa pha, bộ cân bằng hồi tiếp quyết định và hệ thống khôi phục định thời.

12.2.1. Thiết bị phi tuyến băng tần cơ sở

Nếu đặt vào thiết bị phi tuyến băng tần cơ sở là tín hiệu giá trị thực *x(t)*, thì đầu ra của nó *y(t)* cũng là tín hiệu giá trị thực. Thiết bị phi tuyến được mô hình hóa là $y(t) = F\big(x(t)\big)$. Mô hình được dùng phổ biến nhất của thiết bị phi tuyến băng tần cơ sở là mô hình chuỗi mũ và mô hình bộ giới hạn.

Mô hình chuỗi mũ được định nghĩa bởi:

$$y(t) = \sum_{k=0}^{N} a_k x^k(t) \tag{12.7}$$

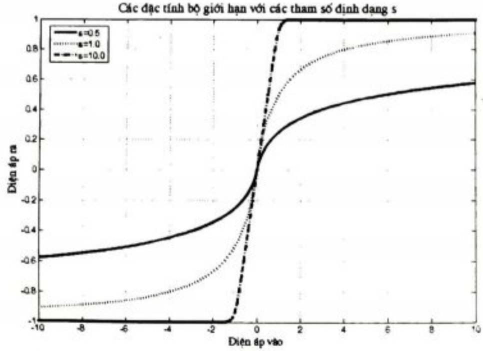
Mô hình bộ giới hạn tổng quát có dạng:

$$y(t) = \frac{M.\text{sgn}\big(x(t)\big)}{\left[1+\left(\frac{m}{|x(t)|}\right)^s\right]^{1/s}} \tag{12.8}$$

Trong phương trình (12.8), M là giá trị giới hạn của đầu ra, m là giá trị giới hạn đầu vào và s là tham số *"định dạng"*. Quan hệ vào/ra chuẩn hóa đối với bộ giới hạn được cho ở hình 12.3 ứng với các giá trị khác nhau của tham số định dạng s. Lưu ý rằng: $s = \infty$ tương ứng với bộ giới hạn "*mềm*", $m = 0$ tương ứng bộ giới hạn "*cứng*". Cũng cần lưu ý thêm, với $m = 0$, thì giá trị của s không ảnh hưởng đến đặc tính của thiết bị phi tuyến được mô tả bởi (12.8). Mã chương trình Matlab tạo ra kết quả hình 12.3 được cho ở file **NVD12_limiter.m** (có trong Phụ lục 12A).

Để được rõ hơn, bạn nên chạy chương trình Matlab với các giá trị khác nhau của s và m và so sánh nhận xét.

Mô phỏng mô hình được cho bởi (12.7) và (12.8) là khá đơn giản, và thường được thực hiện trong miền thời gian. Ta chỉ việc tạo các giá trị mẫu của *x(t)*, xử lý các mẫu này bằng (12.7) hoặc (12.8) để tạo các mẫu của *y(t)*. Sau đó, dùng các mẫu của *y(t)* làm đầu vào cho các khối luồng xuống. Có thể ước tính các thuộc tính của *y(t)* từ các mẫu $\{y(kT_s)\}$ khi *x(t)* là một quá trình ngẫu nhiên. Ta cần phải đặc biệt lưu ý đến tốc độ lấy mẫu như đã đề cập.



Hình 12.3: Các đặc tính bộ giới hạn với các giá trị của tham số định dạng s

12.2.2. Thiết bị phi tuyến thông dải - Mô hình thông dải vùng

Mô hình thông dải (thông băng) không nhớ được dùng để đặc tính hóa cho một lượng lớn thiết bị thông dải phi tuyến băng hẹp trong hệ thống truyền thông. Từ *không nhớ* không những

thể hiện quan hệ vào/ra tức thì mà còn thể hiện thiết bị đó không biểu lộ tính cách chọn lọc tần số trong dải tần hoạt động. Cả độ rộng băng tần của thiết bị phi tuyến và độ rộng băng tần của tín hiệu đều được coi là nhỏ hơn rất nhiều so với tần số sóng mang f_c.

Khi độ rộng băng tần trở nên rộng hơn, thì thiết bị phi tuyến biểu lộ tính cách chọn lọc tần số, và ta dùng các mô hình chọn lọc tần số. *Tính chọn lọc tần số thường đồng nghĩa với tính có nhớ,* và mô hình được dùng phổ biến nhất cho các thiết bị phi tuyến chọn lọc tần số (nghĩa là, thiết bị phi tuyến có nhớ) bao gồm thiết bị phi tuyến không nhớ kẹp giữa 2 bộ lọc như được minh họa ở hình 12.1. Ta sẽ tập trung vào thiết bị phi tuyến thông dải không nhớ và thiết bị phi tuyến thông dải có nhớ ở phần sau.

Xét một thiết bị phi tuyến không nhớ dạng:

$$y(t) = x(t) - 0{,}2x^3(t) \tag{12.9}$$

Giả thiết, đầu vào là một tín hiệu ngẫu nhiên thông dải dạng:

$$x(t) = A(t)\cos\left[2\pi f_c t + \phi(t)\right] \tag{12.10}$$

Trong đó biên độ *A(t)* và độ lệch pha $\phi(t)$ đều là quá trình ngẫu thiên thông thấp có độ rộng băng thông B << f_c (giả định băng hẹp). Từ (12.9) và (12.10) ta được đầu ra *y(t)* của thiết bị phi tuyến là:

$$\begin{aligned} y(t) &= A(t)\cos\left[2\pi f_c t + \phi(t)\right] - 0{,}2\left\{A(t)\cos\left[2\pi f_c t + \phi(t)\right]\right\}^3 \\ &= A(t)\cos\left[2\pi f_c t + \phi(t)\right] - \frac{0{,}2}{4}A^3(t)\left\{\cos\left[6\pi f_c t + 3\phi(t)\right] + 3\cos\left[2\pi f_c t + \phi(t)\right]\right\} \\ &= \left[A(t) - \frac{0{,}6}{4}A^3(t)\right]\cos\left[2\pi f_c t + \phi(t)\right] - \frac{0{,}2}{4}A^3(t)\cos\left[6\pi f_c t + 3\phi(t)\right] \end{aligned} \tag{12.11}$$

Trong phương trình trên, thành phần thứ nhất nằm tại trung tâm (tần số sóng mang f_c) và thành phần cuối cùng là nằm tại thành phần hài thứ 3 của tần số sóng mang $3f_c$. Độ rộng băng thông của thành phần hài thứ ba sẽ là bội của 3B. Do giả thiết $f_c >> B$, nên thành phần thứ hai sẽ nằm ngoài độ rộng băng tần quan tâm. Vì vậy, ta có thể lấy xấp xỉ đầu ra vùng thứ nhất của thiết bị phi tuyến là:

$$z(t) \approx \left[A(t) - \frac{0{,}6}{4}A^3(t)\right].\cos\left[2\pi f_c t + \phi(t)\right] \tag{12.12}$$

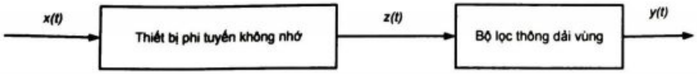
$$z(t) \approx f\left(A(t)\right).\cos\left[2\pi f_c t + \phi(t)\right] \tag{12.13}$$

Trong đó:

$$f\left(A(t)\right) = \left[A(t) - \frac{0{,}6}{4}A^3(t)\right] \tag{12.14}$$

Theo đó, mô hình cho thiết bị phi tuyến băng hẹp không nhớ là một thiết bị phi tuyến không nhớ theo sau là một bộ lọc thông dải "*vùng*" (bộ lọc chỉ cho qua đầu ra "*vùng đầu tiên*" gần f_c). Mô hình được minh họa ở hình 12.4, trong đó *x(t)* và *y(t)* thể hiện đầu vào/ra của mô hình và tần số trung tâm của bộ lọc thông dải vùng là f_c. Thiết bị phi tuyến không nhớ tự nó không có các đáp ứng khác nhau với tín hiệu đầu vào băng tần cơ sở hay thông dải và nó cũng

không nhạy cảm với tần số sóng mang. Chính bộ lọc thông dải vùng đã chuyển mô hình băng tần cơ sở thành mô hình thông dải tại tần số f_c.



Hình 12.4: Mô hình thông dải vùng cho thiết bị phi tuyến băng hẹp không nhớ

Lưu ý rằng, với mô hình chuỗi mũ, đầu ra thông dải *y(t)* có cùng dạng như đầu vào, quan hệ biên độ vào/ra theo hàm *f(A(t))*, pha đầu ra giống pha đầu vào. Hàm *f(A(t))* được xem như đặc tính truyền đạt biên độ-biên độ AM-AM của thiết bị phi tuyến. Một bộ giới hạn hoặc một mô hình chuỗi mũ chỉ ảnh hưởng lên biên độ của tín hiệu vào. Mô hình không tác động vào pha.

Ở dạng đường bao phức của đầu vào *x(t)* và đầu ra *z(t)*, thì mô hình tương đương thông thấp cho thiết bị phi tuyến chuỗi mũ là :

$$\tilde{x}(t) = A(t).e^{j\phi(t)} \tag{12.15}$$

$$\tilde{z}(t) = f\left[A(t)\right].e^{j\phi(t)} \tag{12.16}$$

Mô hình chuỗi mũ có thể được mô phỏng sử dụng mô hình thông dải vùng được cho ở (12.13) và hình 12.4 hoặc mô hình tương đương thông thấp được cho ở (12.16). Việc mô phỏng mô hình thông dải cần phải lấy mẫu tốc độ cao hơn và tính toán phức tạp hơn so với mô hình tương đương thông thấp. May thay, có thể rút ra các mô hình tương đương thông thấp cho hầu hết các thiết bị phi tuyến không nhớ để cho cả phân tích lẫn đo đạc như chỉ ra trong phần dưới.

12.2.3. Mô hình AM-AM và AM-PM đường bao phức thông thấp

Các thiết bị như bộ khuếch đại thông dải, đáp ứng tín hiệu vào thông dải (có phổ được trung tâm tại tần số sóng mang), tạo tín hiệu ra thông dải. Phổ của tín hiệu đầu ra tập trung ở tần số sóng mang, có thể có độ rộng băng lớn hơn so với độ rộng băng của tín hiệu vào. Những thiết bị này có thể được mô hình bằng cách sử dụng biểu diễn đường bao phức của tín hiệu vào/ra.

Giả sử, tín hiệu vào thiết bị phi tuyến thông dải không nhớ dạng:

$$x(t) = A(t)\cos\left(2\pi f_c t + \phi(t)\right) = A\cos(\alpha) \tag{12.17}$$

Trong đó :

$$\alpha = 2\pi f_c t + \phi(t) \tag{12.18}$$

Tín hiệu ra tương ứng $y(t) = F\left(x(t)\right)$ được biểu diễn như sau :

$$y(t) = f\left(A\cos(\alpha)\right) \tag{12.19}$$

Do $A\cos(\alpha)$ tuần hoàn theo α, nên *y(t)* cũng tuần hoàn theo α. Tín hiệu đầu ra *y(t)* có thể được khai triển theo chuỗi Fourier như sau:

$$y(t) = a_0 + \sum_{k=1}^{\infty}\left[a_k\cos(k\alpha) + b_k\sin(k\alpha)\right] \tag{12.20}$$

Trong đó các hệ số chuỗi Fourier a_k và b_k được cho bởi:

$$a_k = \frac{1}{\pi}\int_0^{2\pi} F\left(A\cos(\alpha)\right)\cos(k\alpha)d\alpha \tag{12.21}$$

$$b_k = -\frac{1}{\pi}\int_0^{2\pi} F\left(A\cos(\alpha)\right)\sin(k\alpha)d\alpha \tag{12.22}$$

Đầu ra vùng thứ nhất *z(t)* lân cận f_c được cho bởi thành phần $k = 1$ trong chuỗi Fourier. Cho ta:

$$z(t) = f_1\left(A(t)\right)\cos\left[2\pi f_c t + \phi(t)\right] - f_2\left(A(t)\right)\sin\left[2\pi f_c t + \phi(t)\right] \tag{12.23}$$

Trong đó:

$$f_1\left(A\right) = a_1 = \frac{1}{\pi}\int_0^{2\pi} F\left(A\cos(\alpha)\right)\cos(\alpha)d\alpha \tag{12.24}$$

$$f_2(A) = -b_1 = -\frac{1}{\pi}\int_0^{2\pi} F\left(A\cos(\alpha)\right)\sin(k\alpha)d\alpha \tag{12.25}$$

Hàm $f_1(A)$ đôi khi được gọi là chuyển đổi Chebyshev bậc 1 và hàm phức $\left[f_1(A) - jf_2(A)\right]$ được gọi là hàm mô tả của thiết bị phi tuyến.

Mô hình được cho bởi (12.23) cũng có thể được biểu diễn là:

$$z(t) = f\left(A(t)\right).\cos\left[2\pi f_c t + \phi(t) + g\left(A(t)\right)\right] \tag{12.26}$$

Trong đó:

$$f\left(A(t)\right) = \sqrt{f_1^2(A) + f_2^2(A)} \tag{12.27}$$

$$g\left(A(t)\right) = \arctan\frac{f_2(t)}{f_1(t)} \tag{12.28}$$

Trong hệ tọa độ cực, ta có:

$$f\left(A\right) = e^{jg(A)} = f_1(A) + jf_2(A) \tag{12.29}$$

Các hàm *f(A)* và *g(A)* là các đặc tính truyền đạt (AM-AM) và (AM-PM) của thiết bị phi tuyến.

Mô hình được cho ở (12.26) là sự tổng quát hóa quan hệ vào/ra được cho ở (12.12). Trong khi mô hình được cho ở (12.12) chỉ giải thích cho sự biến đổi biên độ, thì (12.26) cho thấy cả méo biên độ và méo pha do tính phi tuyến, và nó có thể được biểu diễn ở dạng tương đương thông thấp phức của đầu vào/ra như sau:

$$\tilde{x}(t) = A(t)e^{j\phi(t)} \tag{12.30}$$

$$\tilde{z} = f\left(A(t)\right).e^{jg(A)}.e^{j\phi(t)} \tag{12.31}$$

Mô hình được cho trong phương trình trên ở dạng cực và có thể chuyển về dạng:

$$z(t) = s_d(t).\cos\left[2\pi f_c t + \phi(t)\right] - s_q(t).\sin\left[2\pi f_c t + \phi(t)\right] \tag{12.32}$$

Trong đó, đường bao phức có thể được biểu diễn theo thành phần đồng pha $s_d(t)$ và vuông pha $s_q(t)$ như sau:

$$\tilde{z}(t) = \left[s_d(t) + js_q(t)\right].e^{j\phi(t)} \tag{12.33}$$

Trong đó các thành phần đồng pha và vuông pha là:

$$s_d(t) = f\left(A(t)\right).\cos\left(gA(t)\right) \tag{12.34}$$

$$s_q(t) = f\left(A(t)\right).\sin\left(gA(t)\right) \tag{12.35}$$

Rút ra biểu thức giải tích của đặc tính AM-AM và AM-PM

Đặc tính truyền đạt AM-AM, AM-PM có được từ thiết bị phi tuyến thông dải sử dụng (12.23). Đối với thiết bị phi tuyến dạng giới hạn cứng [$m = 0$], ta có:

$$f(A) = \frac{4}{\pi}M \tag{12.36}$$

$$g\left(A\right) = 0 \tag{12.37}$$

Giá trị của *f(A)* chỉ là biên độ của thành phần sin cơ bản (thành phần thứ nhất của đầu ra):

Đối với thiết bị phi tuyến kiểu bộ giới hạn mềm [xem (12.8): $s = \infty$], thì *f(A)* và *g(A)* được cho bởi:

$$\frac{f(A)}{A} = \begin{cases} \dfrac{M}{m}, & A < m \\ \dfrac{2}{M}\left[\arcsin\left(\frac{m}{A}\right) + \left(\frac{m}{A}\right)\left(1 - \frac{m^2}{A^2}\right)^{1/2}\right], & A \geq m \end{cases} \tag{12.38}$$

$$g\left(A\right) = 0 \tag{12.39}$$

Cho thấy bộ giới hạn không gây méo pha.

Với bất cứ thiết bị phi tuyến thông dải không nhớ nào, ta đều có thể rút ra được *f(A)* và *g(A)* theo phép giải tích nếu: ***(i)*** cho trước các đặc tính truyền đạt của thiết bị phi tuyến; ***(ii)*** ước lượng được các tích phân (12.21) và (12.22) ở dạng kín. Trong một số trường hợp, ta có thể trực tiếp rút ra mô hình tương đương thông thấp.

Ví dụ, xét một thiết bị phi tuyến chuỗi mũ dạng:

$$y(t) = \sum_{k=1}^{N} a_k x^k(t) \tag{12.40}$$

Tín hiệu thông dải đầu vào *x(t)* được biểu diễn theo đường bao phức thông thấp của nó là:

$$x(t) = \mathrm{Re}\left\{\tilde{x}(t)\mathrm{e}^{j2\pi f_0 t}\right\} = \frac{1}{2}\left[\tilde{x}(t).\mathrm{e}^{j2\pi f_0 t} + \tilde{x}^*(t).\mathrm{e}^{-j2\pi f_0 t}\right] \tag{12.41}$$

Ở dạng đường bao phức, thành phần mũ thứ *n* của *x(t)* được biểu diễn là:

$$x^n(t) = \frac{1}{2^2}\sum_{k=0}^{n}\binom{n}{k}\left[\tilde{x}(t)\right]^k\left[\tilde{x}^*(t)\right]^{n-k} e^{-j2\pi f_0(2k-n)t} \tag{12.42}$$

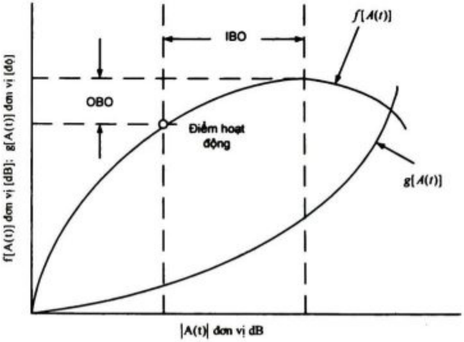
Chỉ các thành phần với n lẻ, và $2k-n=\pm n$ trong $x^n(t)$ mới tạo ra vùng phổ thứ nhất. Do đó, đường bao phức của đầu ra vùng thứ nhất của thiết bị phi tuyến chuỗi mũ là:

$$\tilde{y}(t) = \tilde{x}(t)\sum_{m=0}^{(N-1)/2}\frac{a_{2m+1}}{2^{2m}}.\binom{2m+1}{m+1}.\left|\tilde{x}^*\right|^{2m} \tag{12.43}$$

Phương trình (12.43) mô tả mô hình tương đương thông thấp phức cho loại thiết bị phi tuyến dạng chuỗi mũ [thiết bị phi tuyến chuỗi mũ bậc ba được đề cập ở phần 12.2.3 là trường hợp cụ thể của mô hình được cho bởi (12.43)]. Lưu ý rằng, trong mô hình này không có méo pha nghĩa là $g(A)=0$.

Đo đặc tính AM-AM, AM-PM

Đặc tính AM-AM, AM-PM của bộ khuếch đại thông dải và nhiều thiết bị khác thường được xác định từ đo kiểm và không được rút ra theo cách giải tích. Đầu vào bộ khuếch đại là một sóng mang không điều chế biên độ, biên độ đầu ra $f(A)$ và pha đầu ra $g(A)$ được đo tại các giá trị khác nhau của A. Những phép đo như vậy được gọi là "*phép đo quét công suất*", kết quả được lưu trong bảng dữ liệu của bộ khuếch đại.



Hình 12.5: Đặc tính AM-AM và AM-PM

Đặc tính AM-AM, AM-PM được hiển thị dưới dạng công suất tương đối dB. Chuẩn hóa trục đầu ra theo công suất ra lớn nhất của thiết bị tại điểm bão hòa. Chuẩn hóa trục đầu vào theo công suất đầu vào tại đó tạo ra công suất ra lớn nhất. Những công suất chuẩn hóa này được tham chiếu đến như là độ lùi đầu ra OBO và độ lùi đầu vào IBO. Nếu công suất vào trung bình là rất nhỏ so với công suất vào yêu cầu để tạo ra công suất ra lớn nhất, thì bộ khuếch đại sẽ mang *tính tuyến tính*. Khi công suất vào tăng, thiết bị bắt đầu có tính phi tuyến.

Phép đo thường được thực hiện tại một vài mức công suất đầu vào hay tại các mức của $|A(t)|$; các đặc tính đo được cho vào bảng ứng với các mức công suất vào. Khi mô phỏng có

thể cần đến để nội suy các giá trị trong bảng đó để có được mức công suất đầu vào. Cũng cần lưu ý rằng, mô hình trong (12.30) và (12.31) là cho mức điện áp (dòng điện) của x(t), A(t).v.v... Phép đo AM-AM thường được cho ở dạng các mức công suất. Trong những trường hợp như vậy, cần phải chuyển khuếch đại hay suy hao công suất thành khuếch đại hoặc suy hao điện áp (dòng điện). Minh họa điển hình về các đặc tính AM-AM, AM-PM của bộ khuếch đại thông dải được cho ở hình 12.5.

Dạng giải tích của đặc tính AM-AM và AM-PM

Thực tế, ta thường lấy xấp xỉ đặc tính AM-AM, AM-PM đo bởi dạng giải tích và dùng dạng giải tích để xác định giá trị pha và biên độ đầu ra chứ không sử dụng nội suy. Hai dạng hàm được dùng phổ biến để mô hình hóa các đặc tính đo của bộ khuếch đại RF được cho bởi

$$S_p(A) = \frac{\alpha_p A}{1+\beta_P A^2};\ S_q(A) = \frac{\alpha_q A^3}{\left(1+\beta_q A^2\right)^2} \tag{12.44}$$

$$f(A) = \frac{\alpha_f A}{1+\beta_f A^2};\ \ g(A) = \frac{\alpha_g A^2}{1+\beta_g A^2} \tag{12.45}$$

Tìm được các hệ số của mô hình α_p, α_q, β_p, β_q hoặc α_f, β_f, α_g, β_g từ dữ liệu bằng cách sử dụng kỹ thuật mịn hóa đường cong theo phương pháp số.

Ví dụ 12.1: Mã chương trình Matlab **NVD12_Sim1.m** (có trong Phụ lục 12A) minh họa quá trình tạo các đặc tính AM-AM và AM-PM bằng cách dùng mô hình **Sale'h** như được định nghĩa (12.45). Mã chương trình **NVD12_salehsmodel.m** được viết cho mô hình **Sale'h** được cho trong phụ lục 12A trong đó các tham số của mô hình được xác định là α_f=1,1587; β_f = 1,15; α_g = 4,0; β_g = 2,1.

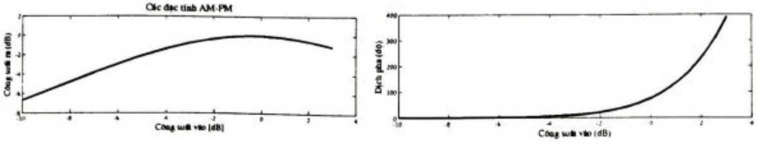
Kết quả chạy chương trình Matlab **NVD12_sim1.m** được cho ở hình 12.6

12.2.4. Mô phỏng mô hình đường bao phức

Mô phỏng các mô hình đường bao phức trong miền thời gian hoặc dưới dạng cực (12.30) và (12.31) hoặc ở dạng cầu phương (12.33); (12.34); (12.35). Sơ khối tương ứng được cho ở hình 12.7. Ở dạng cực, trình tự mô phỏng gồm các bước:

1. Tạo các giá trị mẫu đường bao phức cho đầu vào $\left\{\tilde{x}(kT_s)\right\}$

2. Tính biên độ và pha đầu vào $A(kT_s) = \left|\tilde{x}(kT_s)\right|$ và $\phi(kT_s) = \arg\left(\dfrac{\tilde{x}(kT_s)}{\left|\tilde{x}(kT_s)\right|}\right)$.



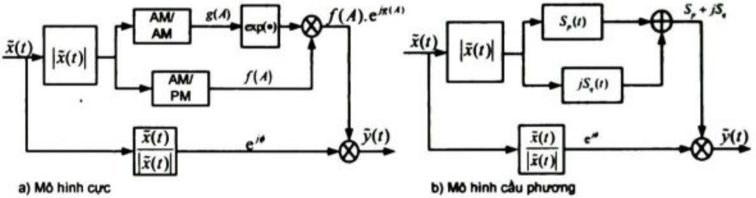
Hình 12.6: Các đặc tính AM-PM

3. Tìm biên độ đầu ra bằng cách dùng đặc tính AM-AM $f(A)$. (Lưu ý: bước này có thể cần phải nội suy các giá trị AM-AM và chuyển khuếch đại công suất thành điện áp hoặc dòng điện).

4. Tìm dịch pha đầu ra bằng cách sử dụng đặc tính AM-PM $g(A)$. (Lưu ý: bước này có thể cần phải nội suy các giá trị AM-PM).

5. Tính các giá trị mẫu của đầu ra phức $\{\tilde{z}(kT_s)\}$ theo (12.31).

Một thủ tục tương tự được dùng để thực hiện mô hình cầu phương như được cho ở hình 12.7. Về mặt tính toán thì hai quy trình là giống nhau.



Hình 12.7: Các mô hình đường bao thông thấp phức của thiết bị phi tuyến thông băng vùng

12.2.5. Trường hợp đa sóng mang

Trong quá trình triển khai mô hình AM-AM, AM-PM, ta đã giả thiết rằng đầu vào thiết bị phi tuyến chỉ có một sóng mang với điều chế biên độ $A(t)$, tần số và pha $\phi(t)$. Trong khi đây là trường hợp hệ thống TDMA đơn sóng mang băng rộng, thì trong hệ thống FDMA tín hiệu vào gồm nhiều sóng mang được điều chế riêng rẽ và tổng các sóng mang này được khuếch đại bởi một bộ khuếch đại công suất. Theo đó, cần phải mở rộng mô hình AM-AM, AM-PM cho trường hợp đa sóng mang

Mô hình đa sóng mang

Giả thiết, đặt vào thiết bị phi tuyến bao gồm tổng m sóng mang điều chế:

$$x(t) = \sum_{k=1}^{m} A_k(t).\cos\left[2\pi\left(f_c + f_k\right)t + \phi_k(t)\right], \quad -\frac{B}{2} \le f_k \le \frac{B}{2} \tag{12.46}$$

Trong đó f_k là lệch tần số của sóng mang thứ k, f_c là tần số trung tâm, $A_k(t)$ và $\phi(t)$ thể hiện điều chế biên độ và pha của sóng mang thứ k. Ta có thể biểu diễn tín hiệu đa sóng mang dưới dạng đường bao phức (xem phần 4.4):

$$x(t) = \mathrm{Re}\left\{\tilde{x}(t)e^{j2\pi f_c t}\right\} \tag{12.47}$$

Trong đó đường bao phức thông thấp $\tilde{x}(t)$ được cho bởi:

$$\tilde{x}(t) = \sum_{k=1}^{m} A_k(t)e^{j2\pi f_k t + j\phi_k(t)} = A(t)e^{j\phi(t)} \tag{12.48}$$

Trong (12.48), biên độ của đường bao phức và pha tương ứng là:

$$A(t) = \sqrt{x_i^2(t) + x_q^2(t)} \tag{12.49}$$

$$\phi(t) = \arctan\left[\frac{x_q(t)}{x_i(t)}\right] \tag{12.50}$$

Trong đó, các thành phần đồng pha và vuông pha tương ứng là:

$$x_i(t) = \sum_{k=1}^{m} A_k(t)\cos\left[2\pi f_k t + \phi_k(t)\right] \tag{12.51}$$

$$x_q(t) = \sum_{k=1}^{m} A_k(t)\sin\left[2\pi f_k t + \phi_k(t)\right] \tag{12.52}$$

Tìm được đường bao phức thông thấp $\tilde{z}(t)$ của đầu ra và tín hiệu ra thông dải *z(t)*:

$$\tilde{z}(t) = f\left(A(t)\right).e^{jg\left(A(t)\right)}.e^{j\phi(t)} \tag{12.53}$$

$$z(t) = \text{Re}\left\{\tilde{z}(t)e^{j2\pi f_c t}\right\} \tag{12.54}$$

Méo điều chế ký sinh trong hệ thống đa sóng mang

Xét một thiết bị phi tuyến dạng chuỗi mũ:

$$y(t) = x(t) - a_3 x^3(t) \tag{12.55}$$

Theo sau là bộ lọc vùng. Giả sử đầu vào thiết bị phi tuyến là tổng của hai đơn sóng mang điều chế

$$x(t) = A_1\cos\left[2\pi(f_c + f_1)t\right] + A_2\cos\left[2\pi(f_c + f_2)t\right] \tag{12.56}$$

Trong đó f_c là tần số sóng mang, f_1 và f_2 là tần số lệch và $f_i < B << f_c$, $i = 1,2$. Có thể thấy rằng, thành phần đầu ra vùng thứ nhất (các thành phần lân cận f_c) được cho bởi:

$$\begin{aligned} z(t) = &\left\{a_1A_1(t) - \left[\tfrac{3}{4}a_3A_1^3(t) + \tfrac{3}{2}a_3A_2^2(t)A_1(t)\right]\right\}\cos\left[2\pi(f_c + f_1)t\right] \\ &+ \left\{a_1A_2(t) - \left[\tfrac{3}{4}a_3A_2^3(t) + \tfrac{3}{2}a_3A_1^2(t)A_2(t)\right]\right\}\cos\left[2\pi(f_c + f_2)t\right] \\ &- \tfrac{3}{4}A_1^2(t)A_2(t)\cos\left[2\pi(f_c + 2f_1 - f_2)t\right] \\ &- \tfrac{3}{4}A_2^2(t)A_1(t)\cos\left[2\pi(f_c + 2f_2 - f_1)t\right] \end{aligned} \tag{12.57}$$

Biểu thức trên cho đầu ra bao gồm các thành phần bị méo tại tần số đầu vào và các thành phần điều chế chéo tại $f_c + 2f_1 - f_2$ & $f_c + 2f_2 - f_1$ (được gọi là các thành phần méo "*điều chế ký sinh*". Các thành phần méo điều chế ký sinh này làm giảm toàn bộ chất lượng tín hiệu và có thể gây ra nhiễu kênh lân cận nếu tần số của những thành phần này làm chúng ở ngoài băng tần tín hiệu xét nhưng lại ở trong băng tần được chiếm giữ bởi tín hiệu lân cận (nhận thấy rằng, thành phần bậc chẵn không tạo ra bất cứ sự méo nào trong vùng lân cận tần số đầu vào).

Cũng có thể dễ dàng thấy rằng, khi đầu vào bao gồm một số lượng lớn sóng mang hoặc khi thiết bị phi tuyến có nhiều thành phần phi tuyến bậc cao hơn, thì đầu ra sẽ chứa một lượng lớn các thành phần kết hợp tuyến tính các tần số đầu vào. Có thể triển khai một thuật toán cho các thành phần điều chế ký sinh và tần số của các thành phần này nhưng rất khó đặc tính hóa và

phân tích ảnh hưởng của méo do tính phi tuyến theo phép giải tích khi đầu vào chứa nhiều sóng mang điều chế hoặc/và tính phi tuyến cao. Tài liệu truyền thông có đủ các kỹ thuật để lấy xấp xỉ ảnh hưởng của méo điều chế ký sinh. Tuy nhiên, việc đặc tính hóa chính xác điều chế ký sinh trong hệ thống số đa sóng mang với các sơ đồ điều chế là khó khăn. Khi này, mô phỏng là rất hữu hiệu.

Ví dụ 12.2: Xét một thiết bị phi tuyến bậc 3 không nhớ dạng

$$y(t) = x(t) - 0,3x^3(t) \tag{12.58}$$

Có một đầu vào thông dải hai đơn sóng mang x(t) tại các tần số 11 Hz và 14 Hz:

$$x(t) = \cos\left[2\pi(11)t\right] + 0,707\cos\left[2\pi(14)t\right] \tag{12.59}$$

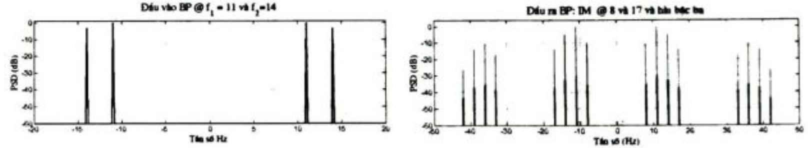
Các phiên bản thông dải của đầu vào/ra được cho ở hình 12.8. Các thành phần điều chế ký sinh được tạo ra bởi thiết bị phi tuyến nằm tại tần số 8 Hz và 17 Hz và thành phần hài bậc 3 xung quanh tần số 33 và 42 Hz.

Phiên bản tương đương thông thấp của ví dụ này, với tần số tham chiếu *f0* = 12 Hz có dạng:

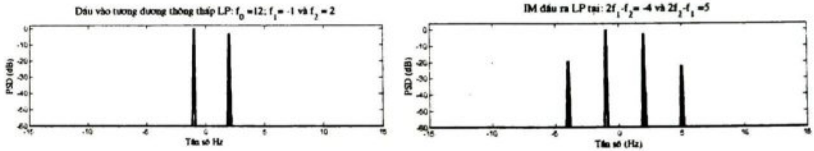
$$\tilde{x}(t) = e^{-j2\pi(1)t} + 0,707\,e^{j2\pi(2)t} \tag{12.60}$$

$$\tilde{y}(t) = \tilde{x}(t) - 0,75(0,3)\left|\tilde{x}(t)\right|^2 \tilde{x}(t) \tag{12.61}$$

Các thành phần điều chế ký sinh bây giờ xuất hiện ở tần số -4Hz và 5 Hz (theo tần số tham chiếu f_0 = 12Hz) như được minh họa ở hình 12.9. Không có thành phần hài bậc 3 được giải thích cho mô hình tương đương thông thấp. Mã chương trình Matlab **NVD12_sim2.m** (có trong Phụ lục 12A) thực hiện bài toán này.



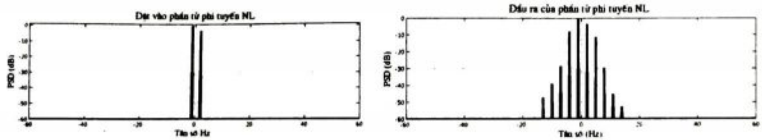
Hình 12.8: Đầu vào/ra của thiết bị phi tuyến dựa trên mô hình thông dải



Hình 12.9: Đầu vào/ra của thiết bị phi tuyến dựa trên mô hình thông thấp

Ví dụ 12.3: Xét thiết bị phi tuyến trên dựa vào đặc tính AM-AM, AM-PM chứ không phải là dự trên mô hình chuỗi mũ như đã làm ở ví dụ trên. Mô hình AM-AM, AM-PM trong trường hợp nàydựa vào mô hình Saleh được định nghĩa thông qua (12.45); Các đặc tính AM-AM, AM-PM được xác định trong ví dụ 12.1, mã chương trình Matlab mô phỏng mô hình Saleh được cho ở phụ lục 12A và chương trình **NVD12_log_psd.m** được cho trong phụ lục 7A

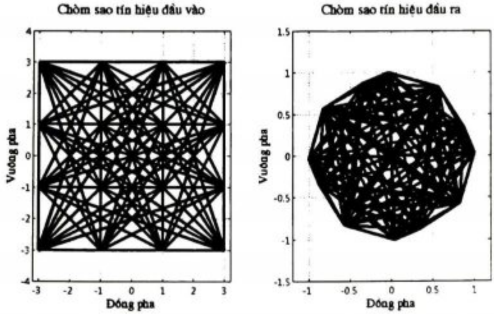
của chương 7. Mô hình tín hiệu tương đương thông thấp là tổng của 2 đơn sóng mang phức trong ví dụ trước được sử dụng để mô phỏng nhằm so sánh kết quả. Chương trình Matlab **NVD12_sim3.m** (có trong phụ lục 12A) thực hiện bài toán này.



Hình 12.10: Các kết quả mô phỏng cho độ lùi 5 dB

Kết quả chạy chương trình Matlab **NVD12_sim3.m** được minh họa ở hình 12.10 với độ lùi 5 dB. (Lưu ý: trong chương trình Matlab ta nhập tham sốđộ lùi là giá trị dương vì được xem là bản chất hơn nhưng cũng có thể chuyển thành giá trị âm trong quá trình gọi mô hình vì mức tín hiệu trong trường hợp này là -5dB so với giá trị đỉnh).

Ví dụ 12.4: Xét một mô hình tín hiệu thông thấp phức cho điều chế 16-QAM. Như trong ví dụ trước sử dụng mô hình Saleh để đặc tính hóa AM-AM và AM-PM. Chòm sao tín hiệu vào/ra được tính toán với độ lùi là 10 dB. Mục đích của mô phỏng là xác định các ảnh hưởng của thiết bị phi tuyến lên tín hiệu. Mã chương trình Matlab **NVD12_sim4.m** (có trong Phụ lục 12A) thực hiện mô phỏng cho bài toán này



Hình 12.11: Chòm sao tín hiệu vào/ra khi độ lùi 10 dB

Kết quả chạy chương trình Matlab **NVD12_sim4.m** được minh họa ở hình 12.11 với độ lùi 10 dB. Lưu ý rằng, ảnh hưởng của tính phi tuyến làm di chuyển nhiều điểm tín hiệu xít lại gần nhau hơn. Như ta đã lưu ý ở chương 4 là: trong môi trường kênh AWGN, thì xác suất lỗi là một hàm đơn điệu của khoảng cách Euclic giữa cặp các điểm trong không gian tín hiệu với xác suất lỗi tăng khi các điểm trong không gian tín hiệu đó gần nhau hơn, hay nói cách khác khi các điểm tín hiệu càng xít gần nhau thì xác xuất lỗi càng tăng và ngược lại. Vì vậy ta kết luận rằng tính phi tuyến làm suy thoái xác suất lỗi của các hệ thống truyền thông. Tương tự với biểu đồ mắt, chòm sao tín hiệu cho ta phép đánh giá *định tính* hiệu năng hệ thống. Theo đó, ta nhìn

nhận sự thay đổi trong chòm sao tín hiệu như là các tham sốbị thay đổi, cho phép ta hiểu sâu về các ảnh hưởng lên hiệu năng hệ thống, đặc biệt có ý nghĩa đối với các hệ thống phi tuyến. Bạn đọc nên thực hiện mô phỏng nhiều lần để quan trắc và so sánh các kết quả khi thay đổi độ lùi.

12.3. Mô hình hóa và mô phỏng thiết bị phi tuyến có nhớ

Nếu đầu ra của một thiết bị phi tuyến phụ thuộc vào giá trị hiện tại và giá trị quá khứ của tín hiệu vào, thì thiết bị đó thuộc loại thiết bị phi tuyến có nhớ. Tính nhớ hay phụ thuộc vào các giá trị trước đó được mô hình hóa trong hệ thống tuyến tính bởi đáp ứng xung kim và tích chập trong miền thời gian. Hệ thống tuyến tính được mô hình hóa trong miền tần số bởi hàm truyền đạt, hàm truyền đạt thể hiện cho đáp ứng của hệ thống trạng thái bền vững phụ thuộc vào tần số đầu vào. ***Vì vậy, tính có nhớ và tính chọn lọc tần số là đồng nghĩa nhau***.

Nhiều thiết bị phi tuyến như các bộ khuếch đại băng rộng biểu lộ tính cách chọn lọc tần số. Tính cách này được thể hiện rõ khi đo đáp ứng của thiết bị tại các mức công suất khác nhau tại các tần số khác nhau. Nếu quan hệ vào/ra không phụ thuộc vào tần số được sử dụng trong phép đo, thì thết bị là thiết bị không nhớ. Ngược lại, thiết bị thể hiện tính cách chọn lọc tần số và do đó có nhớ.

Giả thiết, đưa vào thiết bị AM-AM phi tuyến một đơn mang không điều chế tại một số tần số $f_c + f_i$ với f_c là tần số trung tâm của thiết bị và f_i là tần số lệch so với tần số trung tâm. Đường bao phức của tín hiệu vào là:

$$\tilde{x}(t) = A.\mathrm{e}^{j2\pi f_i t} \tag{12.62}$$

Từ đó:

$$\left|\tilde{x}(t)\right| = A \tag{12.63}$$

Nếu thiết bị phi tuyến không nhớ (không chọn lọc tần số), thì đầu ra được cho bởi:

$$\tilde{y}(t) = f(A)\mathrm{e}^{j2\pi f_i t} \tag{12.64}$$

Lưu ý rằng, biên độ đầu ra độc lập với tần số.

Nếu các phép đo cho thấy, thiết bị phi tuyến thuộc loại chọn lọc tần số, thì phải xét đến tính chọn lọc tần số bằng cách biến đổi hàm AM-AM để chứa tần số đầu vào và viết đáp ứng như sau:

$$\tilde{y}(t) = f\left(AH(f_i)\right).\mathrm{e}^{j2\pi f_i t} \tag{12.65}$$

Khi ta thay đổi f_i trên độ rộng băng tần của thiết bị, thì hàm $H(f_i)$ sẽ giải thích cho tính phụ thuộc tần số của thiết bị phi tuyến. Lưu ý, $H(f_i)$ được xem là hàm truyền đạt của bộ lọc. Bộ lọc tạo ra đáp ứng $A.H(f_i).e^{j2\pi f_i t}$ khi đầu vào là $\tilde{x}(t) = A.e^{j2\pi f_i t}$. Theo đó, mô hình cho phần AM-AM của thiết bị phi tuyến chọn lọc tần số bao gồm một bộ lọc $H(f_i)$ theo sau bởi một thiết bị phi tuyến không nhớ AM-AM, $f(A)$.

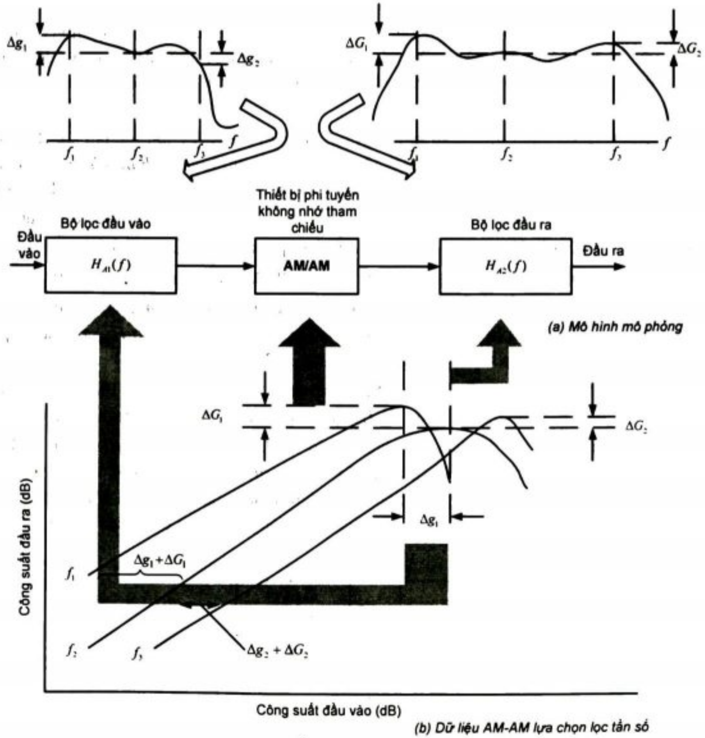
Giải pháp này được mở rộng để giải thích cho các đặc tính truyền đạt AM-AM chọn lọc tần số bằng cách đưa thêm bộ lọc khác vào trong mô hình. Thực vậy, mô hình được dùng phổ biến nhất cho các thiết bị phi tuyến chọn lọc tần số gồm một thiết bị phi tuyến không nhớ kẹp giữa 2 bộ lọc. Ta xét hai mô hình này.

12.3.1. Các mô hình thực nghiệm dựa trên đo kiểm

Các mô hình này rút ra từ các phép đo "*công suất bước*" và "*tone quét*" được thực hiện với tín hiệu vào là một tone (hay đơn sóng mang) không điều chế dạng:

$$x(t) = A_i \cos\left[2\pi(f_i + f_c)t\right] \tag{12.66}$$

Đo độ lệch pha và biên độ đầu ra tại các giá trị khác nhau của tần số f_i và biên độ đầu vào A_i tạo ra một tập các đồ thị như được cho ở hình 12.12(b). Lưu ý rằng, các đồ thị này cho thấy một cách rõ ràng bản chất phụ thuộc vào tần số của đáp ứng của các thiết bị phi tuyến.



Hình 12.12: Minh họa mô hình AM-AM lựa chọn tần số

Hai mô hình được dùng để giải thích cho tính phụ thuộc tần số của thiết bị phi tuyến là mô hình của Paza và mô hình của Saleh. Cả hai mô hình này đều cố gắng tạo ra các phép đo tone quét và công suất bước chính xác nhất ở mức có thể.

Mô hình Poza

Mô hình mô phỏng đơn giản đặc tính hóa các phép đo giống như hình 12.12(b) được rút ra bởi Poza trên cơ sở các giả định sau:

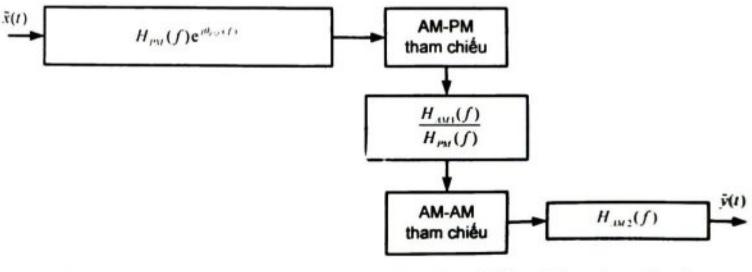
1. **AM-AM:** Các đường cong AM-AM đối với các tần số khác nhau có dạng giống nhau, mọi đường cong sẽ là sự kết hợp của các chuyển dịch theo trục đứng và trục ngang của các đường cong khác.

2. **AM-PM:** Các đường cong AM-PM này đối với các tần số khác nhau cũng có dạng giống nhau, mỗi đường cong là kết hợp của sự dịch theo trục đứng và trục ngang của các đường cong khác.

Mô hình này yêu cầu một thủ tục phù hợp hóa đường cong để làm phù hợp họ các đường cong đối với dữ liệu AM-AM, AM-PM sao cho mỗi phần tử của họ đường cong đó là một phiên bản được dịch của phần tử khác thuộc họ đường cong đó (lưu ý chuyển dịch cả trục đứng và ngang).

Trước hết, ta xét cách thực hiện mô hình mô phỏng AM-AM chọn lọc tần số. Bước đầu tiên là chọn một đường cong trong họ AM-AM làm thiết bị phi tuyến "***tham chiếu***" tại tần số tham chiếu f_c. Theo đường cong này, có thể tìm được đáp ứng AM-AM tại tần số f_1 bằng cách dịch chuyển theo chiều ngang $\Delta G_1 + \Delta g_1$. Việc dịch chuyển theo chiều ngang tương ứng với việc khuếch đại (hoặc suy hao) *tín hiệu vào* của thiết bị phi tuyến chuẩn tại tần số f_1 và sự dịch chuyển theo chiều dọc ΔG_1 tương ứng với khuếch đại (hoặc suy hao) *đầu ra* của thiết bị phi tuyến tiêu chuẩn tại tần số f_1. Những điều này có thể thực hiện được bởi 2 bộ lọc FIR, một ở trước và một ở sau thiết bị phi tuyến tham chiếu có các đáp ứng biên độ $H_{AM1}(f)$ và $H_{AM2}(f)$. Theo đó, mô hình AM-AM có thể thực hiện như hình 12.12.

Mô hình tương tự và việc thực thi có thể được rút ra cho đáp ứng AM-PM. Mô hình này gồm một bộ lọc đầu vào có đáp ứng biên độ $H_{PM}(f)$ ở trước thiết bị phi tuyến tham chiếu AM-PM và một đáp ứng pha $e^{j\theta_{PM}(f)}$ theo sau đầu ra. Có thể kết hợp mô hình AM-AM, AM-PM thành một mô hình như hình 12.13. Phần AM-PM ở phía trước phần AM-AM, và độ lệch (biên độ) đầu vào do mô hình AM-PM gây ra phải được "làm mất" trước mô hình AM-AM sao cho mức công suất đưa vào phần AM-AM của mô hình là giống với mức công suất ở tín hiệu đầu vào $\tilde{x}(t)$.



Hình 12.13: Mô hình AM-AM và AM-PM (được kết hợp) lựa chọn tần số

Mô hình Saleh

Một giải pháp hơi khác để mô hình hóa các thiết bị phi tuyến có nhớ được Saleh đề xuất. Mô hình này đạt được từ mô hình cầu phương không nhớ được cho trong (12.44) và (12.45) bằng cách làm cho các hệ số phụ thuộc tần số. Ta có:

$$S_d(A) = \frac{\alpha_p(f)A}{1+\beta_p(f)A^2} \tag{12.67}$$

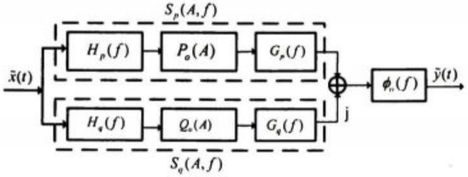
$$S_q(A) = \frac{\alpha_q(f)A^3}{\left[1+\beta_q(f)A^2\right]^2} \tag{12.68}$$

Tìm được các hệ số từ các phép đo tại các tần số khác nhau bằng cách sử dụng phù hợp hóa bình phương nhỏ nhất. Việc thực thi thực tế mô hình được cho ở hình 12.14. Các hàm được minh họa trong hình 12.14 được định nghĩa như sau: $\phi_0(f)$ là đáp ứng pha tín hiệu nhỏ.

$$H_P(f)=\sqrt{\beta_P(f)};\quad H_q(f)=\sqrt{\beta_q(f)};\quad P_0=A/(1+A^2);\quad G_P(f)=\alpha_P(f)/\sqrt{\beta_P(f)};$$

$$G_q(f)=\alpha_q(f)/\sqrt{\beta_q^{3/2}(f)};\qquad Q_0(A)=A^3/\left[1+A^2\right]^2 T$$

Lưu ý rằng, mô hình Saleh và mô hình Poza bắt đầu với các giả định khác nhau và có cấu hình khác nhau. Chúng đều có dạng " mô hình sơ đồ khối" được thiết kế để tái tạo một tập các phép đo cụ thể. Cả hai mô hình này đều có cấu trúc phức tạp hơn các mô hình thiết bị phi tuyến không nhớ. Sự tăng tính phức tạp là do dùng các thủ tục có tính kinh nghiệm để làm phù hợp các tham sốmô hình với dữ liệu đo được.



Hình 12.14: Mô hình cầu phương của Saleh cho thiêt bị phi tuyến có nhớ

12.3.2. Các mô hình khác

Mặc dù phức tạp, nhưng các mô hình được mô tả trong phần trước không có khả năng bắt giữ tính cách của thiết bị phi tuyến khi đầu vào chứa tổng của nhiều sóng mang điều chế, bởi lẽ chúng được rút ra dựa trên các phép đo 1 sóng mang đường bao không đổi. Do tính chất xếp chồng không áp dụng được cho hệ thống phi tuyến, nên không thể đặc tính hóa tính cách với đầu vào nhiều sóng mang dựa trên các phép đo một tần số. Dĩ nhiên, thật khó để tạo ra các phép đo đa sóng mang cho nhiều kết hợp của các mức công suất và các tần số đầu vào. Số lượng các kết hợp này sẽ quá lớn khi số các sóng mang nhiều.

Một vài giải pháp khác, một trong số đó là xấp xỉ tính cách của thiết bị phi tuyến sử dụng một sóng mang điều chế với chuỗi giả ngẫu nhiên PN làm đầu vào. Đầu vào này xấp xỉ như

một tổng của số lượng lớn các sóng mang có mật độ phổ công suất đồng đều ít hoặc đồng điều nhiều (lưu ý rằng, bằng giải pháp này, thì các thành phần phổ đầu vào được tạo ra bởi tín hiệu điều chế). Bằng cách thay đổi công suất tại đầu vào, dùng các phép đo để đặc tính hóa tính cách của thiết bị phi tuyến dưới điều kiện đầu vào (xấp xỉ như trường hợp đầu vào đa sóng mang). Mô hình kết quả có hàm truyền đạt dạng $H(f,p)$ trong đó P là mức công suất vào.

Một giải pháp thông dụng hơn đã được đề xuất để mô hình hóa và phân tích thiết bị phi tuyến có nhớ là sử dụng chuỗi Volterra. Quan hệ vào/ra trong miền thời gian được cho bởi:

$$y(t) = \sum_{k=1}^{\infty} y_k(t) \tag{12.69}$$

Trong đó:

$$y_k(t) = \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} h(\tau_1, \dots, \tau_k) x(t-\tau_1) \dots x(t-\tau_k) d\tau_1 \dots d\tau_k \tag{12.70}$$

Mô hình đáp ứng trong miền thời gian được cho trong (12.70) giả thiết rằng hệ thống không có đáp ứng với thành phần một chiều hoặc không đổi trong đầu ra. Lưu ý rằng, mô hình giống với mô hình chuỗi mũ. Tuy nhiên, mỗi thành phần trong chuỗi Volterra là tích chập k lần (không phải là mũ thứ k) tín hiệu vào với đáp ứng xung $h(\tau_1, \tau_2, \dots \tau_k)$ k lần. Có thể rút ra phiên bản tương đương thông thấp của mô hình nhưng rất phức tạp.

12.4. Kỹ thuật giải phương trình vi phân phi tuyến NLDE

Ta biết rằng, có thể mô hình hóa hệ thống phi tuyến có nhớ bằng phương trình vi phân phi tuyến NLDE, và có thể mô phỏng hệ thống bằng cách thực hiện tích phân rời rạc thời gian thay cho tích phân tiên tục thời gian. Ta cũng biết rằng, 2 giải pháp mô phỏng hệ thống được mô tả bởi phương trình vi phân: ***(i)*** Phương pháp *sơ đồ khối*, trong đó sơ đồ khối hệ thống được xây dựng bằng các khối cơ bản gồm một vài các khối phi tuyến không nhớ đơn giản hơn, các bộ tích phân, bằng cách mô phỏng mô hình sơ đồ khối của hệ thống phi tuyến đó tìm được nghiệm; ***(ii)*** Rút ra phương trình vi phân phi tuyến kiểm soát tính cách động của PLL và giải NLDE bằng cách sử dụng các thủ tục đệ quy, phương pháp này được hiểu như phương pháp *trực tiếp*, sẽ chính xác và ổn định hơn trong thời gian một mẫu cho trước. Tuy nhiên, cần có một số nỗ lực để triển khai mô hình và thực hiện giải pháp này, và các nỗ lực đó phải được lặp lại cho mỗi hệ thống phi tuyến mới.

Mọi giải pháp đều dẫn đến cùng kết quả khi thời gian mẫu hay thời gian bước mô phỏng là nhỏ. Tuy nhiên, nếu ta muốn bước thời gian lớn để giảm tải tính toán, thì mô hình trực tiếp sử dụng kỹ thuật nghiệm kích cỡ bước khả biến sẽ rất hiệu quả. Kích cỡ bước thường được chọn tự động bởi phương pháp nghiệm tùy vào tính cách của NLDE cơ bản. Trong các vùng nơi nghiệm thể hiện tốt, ta có thể dùng bước thời gian lớn để tiết kiệm thời gian mô phỏng. Tuy nhiên, cần một vài phép nội suy nếu các khối kế tiếp nhau yêu cầu các mẫu cách đều nhau.

Khi vòng khóa pha PLL được mô phỏng bởi chính nó, yêu cầu về bước thời gian sẽ được xác định bởi "độ rộng băng thông của vòng ". Nếu PLL được mô phỏng cùng các thành phần khác của máy thu, thì tốc độ lấy mẫu cho phần còn lại của máy thu sẽ được kiểm soát bởi tốc độ dữ liệu R. Ở dạng tổng quát, R sẽ lớn hơn rất nhiều so với độ rộng băng thông của vòng và do đó bước thời gian tương xứng với tốc độ dữ liệu R sẽ nhỏ hơn rất nhiều bước thời gian để mô

phỏng chính xác chính PLL đó. Khi này, ta nên sử dụng mô phỏng đa tốc độ, trong đó dùng các bước thời gian khác nhau cho các phần khác nhau của hệ thống.

Mục đích của phần này là mô phỏng hệ thống phi tuyến có nhớ sử dụng nghiệm đệ qui của NLDE cơ bản. Ta sử dụng PLL đã được đề cập ở chương 6 làm ví dụ để minh họa cho phương pháp luận.

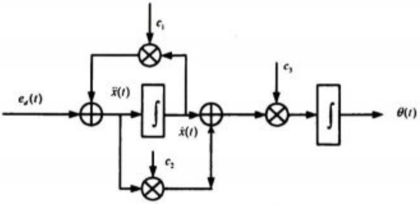
12.4.1. Dạng vectơ trạng thái của NLDE

Tồn tại nhiều tài liệu về phương pháp số để giải phương trình vi phân phi tuyến. Từ quan điểm mô phỏng, các kỹ thuật được thực hiện ở dạng đệ qui là hấp dẫn nhất trên cả hai phương diện cấu trúc và tính toán. Với phương trình vi phân phi tuyến bậc *m*, nghiệm đệ qui trong miền thời gian thường dẫn đến sử dụng phương pháp biến trạng thái trong đó các phương trình hệ thống được biểu diễn ở dạng

$$\dot{\mathbf{X}}(t) = \mathbf{AX}(t) + \mathbf{BU}(t) \tag{12.71}$$

$$\mathbf{Y}(t) = \mathbf{DX}(t) \tag{12.72}$$

Trong đó X là vectơ trạng thái kích thước *mx1*, U &Y là vectơ vào/ra tương ứng, A, B, D là các ma trận hằng số định nghĩa hệ thống, và m là bậc của hệ thống.



Hình 12.15: Biểu diễn bộ lọc vòng và định nghĩa các biến trạng thái cho mô hình NLDE của PLL

Để áp dụng kỹ thuật này cho PLL, xét quan hệ giữa $e_d(t)$ và pha đầu ra $\theta(t)$ của VCO (như được cho ở hình 12.15). Đây là phần tuyến tính của hệ thống, từ hình 12.15 quan hệ giữa $e_d(t)$ và $\theta(t)$ được xác định bởi:

$$\theta(t) = \int c_3\left[\dot{x}(t) + c_2\ddot{x}(t)\right]dt = c_3x(t) + c_2c_3\dot{x}(t) \tag{12.73}$$

$$\ddot{x}(t) = e_d(t) + c_1\dot{x}(t) \tag{12.74}$$

Từ định nghĩa về $e_d(t)$ và (12.73):

$$e_d(t) = \sin\left[\phi(t) - \theta(t)\right] = \sin\left[\phi(t) - c_2c_3\dot{x}(t) - c_3x(t)\right] \tag{12.75}$$

Thay vào (12.74) cho ta phương trình vi phân:

$$\ddot{x}(t) = \sin\left[\phi(t) - c_2c_3\dot{x}(t) - c_3x(t)\right] + c_1\dot{x}(t) \tag{12.76}$$

Từ hình 6.3 và (6.25) [*xem chương 6*] ta biết rằng:

$$\Theta(s) = G\frac{1}{s}\frac{s+a}{s+\lambda a}E_d(s) \tag{12.77}$$

Và từ hình 12.15 ta có:

$$\Theta(s) = \frac{c_3}{s}\frac{sc_2+1}{s-c_1}E_d(s) = c_2c_3\frac{1}{s}\frac{s+\frac{2}{c_2}}{s-c_1}E_d(s) \tag{12.78}$$

Từ đó, nhận được:

$$c_1 = -\lambda a, \quad c_2 = \frac{1}{a}, \quad c_2c_3 = G, \quad c_3 = aG \tag{12.79}$$

Tại đây, nếu ta định nghĩa các biến trạng thái là:

$$\begin{aligned} x_1(t) &= x(t) \\ x_2(t) &= \dot{x}_1(t) \end{aligned} \tag{12.80}$$

Từ (12.76), ta có thể viết phương trình vi phân phi tuyến NLDE ở dạng ma trận:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ \sin\left[\phi(t) - c_2c_3x_2(t) - c_3x_1(t)\right] + c_1x_2(t) \end{bmatrix} = \begin{bmatrix} f_1(x_1,x_2,\phi) \\ f_2(x_1,x_2,\phi) \end{bmatrix} \tag{12.81}$$

Lưu ý rằng, NDLE cũng có thể được biểu diễn là:

$$\dot{\mathbf{X}} = F(\mathbf{X}, t, \mathbf{U}) \tag{12.82}$$

Trong đó X là vectơ kích cỡ m, và F là một hàm vectơ phi tuyến với m thành phần và U là vectơ đầu vào mà trong ví dụ này là vô hướng $\phi(t)$.

Tổng quát, ta có thể chuyển NLDE bậc m thành một tập m phương trình vi phân phi tuyến NLDE bậc 1 đồng thời, nó có dạng giống với dạng biến trạng thái của các phương trình vi phân tuyến tính bậc m. Đến đây, ta xét các phương pháp giải các phương trình vi phân bậc một đồng thời (nghĩa là vectơ), bắt đầu bằng phương trình vi phân bậc một vô hướng.

12.4.2. Nghiệm đệ qui của NLDE - Trường hợp vô hướng

Các kỹ thuật hiện

Xét phương trình vi phân bậc một dạng:

$$\dot{x}(t) = f(x, t, u) \tag{12.83}$$

Với điều kiện đầu $x(t_0)$, x_n là nghiệm đạt được bằng tích phân số tại bước thời gian t_n (ta sẽ dùng ký hiệu $x(t_n)$ là nghiệm chính xác là ẩn số chưa biết, và dùng ký hiệu x_n là nghiệm xấp xỉ đạt được bằng cách tích phân số).

Hầu hết các phương pháp tích phân số để giải các phương trình vi phân phi tuyến (hoặc tuyến tính) đều dựa trên khai triển chuỗi Taylor. Ví dụ, xét khai triển chuỗi Taylor (hoặc một số các chuỗi khác) dạng:

$$x(t_{n+1}) = x(t_n + h_n) = x(t_n) + h_n\dot{x}(t_n) + T(h_n) \tag{12.84}$$

Tong đó h_n là bước thời gian $t_{n+1}-t_n$, và T là lỗi nội bộ hay phần dư được cho bởi:

$$T(t_n) = h_n^2\ddot{x}(\xi)/2, \quad t_n \le \xi \le t_{n+1} \tag{12.85}$$

Nếu h_n đủ nhỏ, thì ta có thể đạt được một nghiệm đệ qui cho phương trình vi sai bậc 1:

$$x_{n+1} = x_n + h_n\dot{x}_n = x_n + h_n f_n \tag{12.86}$$

$$f_n = f\left(x_n, t_n, u(t_n)\right) \tag{12.87}$$

Đệ qui có thể được bắt đầu với điều kiện đầu $x(t_0) = x_0$. Có thể rút ra một phiên bản đơn giản hơn của nghiệm hồi qui với kích thước bước thời gian cố định h là:

$$x_{n+1} = x_n + h\dot{x}_n = x_n + hf\left(x_n, t_{n-1}, u(t_n)\right) \tag{12.88}$$

$$t_{n+1} = t_n + h \tag{12.89}$$

Các phương trình (12.88) và (12.89) định nghĩa phương pháp tích phân Euler.

Bằng cách tính đến các đạo hàm bậc cao hơn trong chuỗi Taylor, ta có thể rút ra được các công thức tính tích phân chính xác hơn. Ví dụ, tích phân "*bậc 2*" có thể được rút ra bằng cách bắt đầu từ:

$$x_{n+1} = x_n + h\dot{x}_n + \frac{1}{2}h^2\ddot{x}_n \tag{12.90}$$

Sử dụng phép xấp xỉ cho đạo hàm bậc 2:

$$\ddot{x}_n \approx \frac{\dot{x}_n - \dot{x}_{n-1}}{h} = \frac{f\left(x_n, t_n, u(t_n)\right) - f\left(x_{n-1}, t_{n-1}, u(t_{n-1})\right)}{h} = \frac{f_n - f_{n-1}}{h} \tag{12.91}$$

Ta được tích phân Adam—Bashforth bậc 2:

$$x_{n+1} = x_n + hf_n + h^2\frac{f_n - f_{n-1}}{2h} = x_n + \frac{h\left(3f_n - f_{n-1}\right)}{2} \tag{12.92}$$

Một lớp các phép tích phân tương tự là các phương pháp Runge-Kutta. Một trong những công thức được sử dụng rộng rãi nhất của R-K là công thức bốn tầng kinh điển:

$$x_{n+1} = x_n + \frac{h\left(k_1 + k_2 + k_3 + k_4\right)}{6} \tag{12.93}$$

Trong đó:

$$k_1 = f\left(x_n, t_n, u(t_n)\right) \tag{12.94}$$

$$k_2 = f\left(x_n + \frac{hk_1}{2}, t_n + \frac{h}{2}, u\left(t_n + \frac{h}{2}\right)\right) \tag{12.95}$$

$$k_3 = f\left(x_n + \frac{hk_2}{2}, t_n + \frac{h}{2}, u\left(t_n + \frac{h}{2}\right)\right) \tag{12.96}$$

$$k_4 = f\left(x_n + \frac{hk_3}{2}, t_n + \frac{h}{2}, u\left(t_n + \frac{h}{2}\right)\right) \tag{12.97}$$

Cả ba phương pháp mô tả trên đều được gọi là các phương pháp hiện, vì đệ qui được thực hiện một cách hiện, sử dụng các giá trị nghiệm và giá trị đạo hàm đạt được tại các bước thời gian trước.

Các kỹ thuật ẩn

Một lớp các kỹ thuật nghiệm khác được gọi là các kỹ thuật ẩn cho ta các nghiệm chính xác và ổn định hơn nhưng lại tăng khối lượng tính toán. Trong các kỹ thuật ẩn, nghiệm tại bước thời gian t_{n+1} không chỉ bao gồm các đại lượng được tính toán ở bước thời gian trước t_n mà còn bao gồm các đại lượng được tính toán tại bước thời gian hiện hành t_{n+1}. Điều này cần có nghiệm của một phương trình đại số phi tuyến tại mỗi bước thời gian.

Một kỹ thuật ẩn rất phổ biến và rất đơn giản là phép tích phân hình thang được cho bởi:

$$x_{n+1} = x_n + \frac{h}{2}\left[\dot{x}_n + \dot{x}_{n+1}\right] = x_n + \frac{h}{2}\left[f\left(x_n, t_n, u(t_n)\right) + f\left(x_{n+1}, t_{n+1}, u(t_{n+1})\right)\right] \tag{12.98}$$

Đây chỉ là một sự xấp xỉ các diện tích hình thang trong tổng Reiman của tích phân (ta đã đề cập tích phân này ở chương 5). Thấy rõ từ phương trình trên, x_{n+1} đều xuất hiện ở cả 2 vế của phương trình thể hiện tính ẩn và ta phải giải phương trình phi tuyến (12.98) để tìm x_{n+1}, trong khi đó với phương pháp hiện x_{n+1} không xuất hiện trong hàm phi tuyến $f(.)$ ở vế phải, và vì vậy ta có thể thấy nghiệm.

Tồn tại nhiều phương pháp ẩn khác ngoài phương pháp hình thang. Ví dụ về kỹ thuật ẩn, ta xét phương pháp Adam-Moulton bậc 3 trên cơ sở khai triển chuỗi Taylor:

$$x(t_{n+1}) \approx x_n + h\dot{x}_n + \frac{h^2\ddot{x}_n}{2} + \frac{h^3 x_n}{6} \tag{12.99}$$

Với phép xấp xỉ đối xứng cho các đạo hàm:

$$\dot{x}_n = f\left(x_n, t_n, u(t_n)\right) = f_n \tag{12.100}$$

$$\ddot{x}_n = \frac{\dot{x}_{n+1} - \dot{x}_{n-1}}{2h} \tag{12.101}$$

$$x_n = \frac{\dot{x}_{n+1} - 2\dot{x}_n + x_{n-1}}{h^2} \tag{12.102}$$

Thế các phương trình này vào (12.99), ta được công thức tích phân Adam-Moulton "*hai bước*":

$$x_{n+1} = x_n + \frac{h\left(5f_{n+1} + 8f_n - f_{n-1}\right)}{12} \tag{12.103}$$

Công thức tích phân A-M trên cho thấy hai vấn đề chính với các kỹ thuật ẩn bậc cao hơn là: điều kiện đầu, và giải phương trình phi tuyến tại mỗi bước thời gian Theo đó: *(i)* Khi áp dụng đệ qui được cho bởi (12.103) để tìm x_1 ta cần có x_0 và x_{-1}. Trong khi điều kiện đầu x_o đã được cho, thì x_{-1} thường không có sẵn. Để tránh vấn đề này, ta bắt đầu lặp tại $n+1 = 2$, bằng cách dùng giá trị của x_1, nó có thể được tính toán dùng phương pháp hiện như phương pháp R-K; *(ii)* Xuất hiện khó khăn thứ 2 vì x_{n+1} xuất hiện ở vế phải bên trong hàm (phi tuyến) f, vì vậy

nghiệm x_{n+1} sẽ yêu cầu nghiệm số của phương trình ẩn (12.103). Ta trình bày hai phương pháp lặp để giải quyết vấn đề này.

Nghiệm ẩn sử dụng phương pháp dự đoán-hiệu chỉnh

Ý tưởng của phương pháp dự đoán-hiệu chỉnh (P-C) là: ***(i)*** *Trước tiên* tìm một giá trị dự đoán cho x_{n+1} bằng kỹ hiện như phượng pháp R-K hay phương pháp A-B (cùng bậc như phương pháp A-M) và sử dụng kết quả dự đoán (ước tính) của x_{n+1} này vào vế phải phương trình (12.103) tìm giá trị hiệu chỉnh (được cải thiện) của x_{n+1}; ***(ii)*** *Sau đó* ta lại tiến hành với $n + 2$. Để cải thiện độ chính xác của nghiệm, ta có thể sử dụng kỹ thuật lặp tại mỗi bước thời gian và lặp lại phương pháp dự đoán-hiệu chỉnh nhiều lần. Tại đây, (12.103) được biến đối thành:

$$x_{n+1}^{r+1} = x_n + \frac{h(5f_{n+1}^r + 8f_n - f_{n-1})}{12}, r = 1,2,3,\ldots \tag{12.104}$$

Trong đó, r là chỉ số lặp. Lặp được bắt đầu bằng việc sử dụng giá trị dự đoán x^1_{n+1} của x_{n+1} (đạt được thông qua phương pháp hiện) trong vế phải của (12.104) để tạo giá trị tiếp theo trong bước lặp x^2_{n+1}, sau đó giá trị này lại được sử dụng cho vế phải của (12.104) để đạt được giá trị được cải thiện tiếp theo và cứ như vậy cho đến khi lặp hội tụ. [Lưu ý rằng, x_{n+1}^{r+1} là giá trị của x_{n+1} được lặp lần thứ *(r+1)* chứ không phải là số mũ *(r+1)* của x_{n+1}].

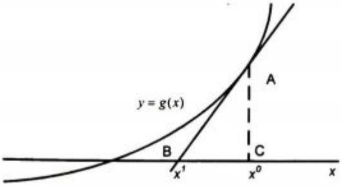
Nghiệm ẩn sử dụng phương pháp Newton-Raphson

Nhiều kỹ thuật khác như Newton-Raphson (N-R) để giải phương trình (12.103). Phương pháp N-R dựa vào kỹ thuật lặp để tìm nghiệm $y = g(x) = 0$, nghĩa là tìm giá trị của x sao cho $g(x) = 0$

Giải pháp lặp cho bài toán "tìm nghiệm" như sau:

Tham chiếu đến hình 12.16, cắt đường cong $y = g(x)$ tại x^0 được cho bởi:

$$\dot{g}(x^0) = \frac{AC}{BC} = \frac{g(x^0)}{x^0 - x^1} \tag{12.105}$$



Hình 12.16: Kỹ thuật lặp để giải $g(x) = 0$

Sắp xếp lại phương trình trên thành:

$$x^1 = x^0 - \frac{g(x^0)}{\dot{g}(x^0)} \tag{12.106}$$

Nếu quá trình này được lặp để tạo ra chuỗi các giá trị x^1, x^2, x^3 , bằng quan hệ truy toán:

$$x^{r+1} = x^r - \frac{g(x^r)}{\dot{g}(x^r)} \tag{12.107}$$

Thì x^r sẽ hội tụ đến giá trị 0 của hàm *g(x)* trong các điều kiện khá đơn giản.

Kỹ thuật này có thể được áp dụng để giải (12.103) bằng cách trước hết viết nó như sau:

$$g(x_{n+1}) = x_{n+1} - x_n - \frac{h(5f_{n+1} + 8f_n - f_{n-1})}{12} = 0 \tag{12.108}$$

Và ứng dụng đệ quy được cho trong (12.107):

$$x_{n+1}^{r+1} = x_{n+1}^r - \frac{g(x_{n+1}^r)}{\dot{g}(x_{n+1}^r)} \tag{12.109}$$

Quá trình lặp cho đến khi sự khác nhau giữa các giá trị liền kề là nhỏ. Tìm giá trị bắt đầu $x^1{}_{n+1}$ của x_{n+1} bằng phương pháp hiện.

12.4.3. Dạng tổng quát của phương pháp đa bước

Dạng nghiệm đệ qui tổng quát của phương trình vi phân phi tuyến NLDE được biểu diễn là:

$$x_{n+1} = \sum_{i=0}^{p} a_i x_{n-i} + h \sum_{j=-1}^{p} b_j f_{n-j} \tag{12.110}$$

được xem là tích phân p bước và nó có thể thuộc loại hiện hoặc ẩn tùy thuộc vào $b_{-1} = 0$ (hiện) hay $b_{-1} \neq 0$ (ẩn). Bảng 12.1 tóm tắt các công thức tích phân khác nhau thường được dùng trong mô phỏng. Tìm được lỗi cắt được cho trong bảng từ lỗi cắt liên quan đến việc kết thúc khai triển chuỗi Taylor.

Bảng 12.1: Các hàm tích phân thường được dùng trong mô phỏng

Phương pháp	Các hệ số	Lỗi cắt xén
Euler tiến	$p = 0; a_0 = 1; b_{-1} = 0; b_0 = 1$ $x_{n+1} = x_n + hf_n$	$\frac{1}{2}h^2 \ddot{x}(\xi), \quad t_n \le \xi \le t_n + h$
Euler lùi	$p = 0; a_0 = 1; b_{-1} = 0; b_0 = 1$ $x_{n+1} = x_n + hf_{n+1}$	$\frac{1}{2}h^2 \ddot{x}(\xi), \quad t_n \le \xi \le t_n + h$
Hình thang	$p = 0; a_0 = 1; b_{-1} = \frac{1}{2}; b_0 = \frac{1}{2}$ $x_{n+1} = x_n + \frac{h}{2}[f_n + f_{n+1}]$	$\frac{1}{12}h^3 \dddot{x}(\xi)$
Adam-Bash-2	$p = 0; a_0 = 1; b_{-1} = 0; b_0 = \frac{3}{2}; b_1 = -\frac{1}{2}$ $x_{n+1} = x_n + \frac{h}{2}[3f_n - f_{n+1}]$	$\frac{5}{12}h^3 \dddot{x}(\xi)$
Adam-Moulton-2	$p = 1; a_0 = 1; b_{-1} = \frac{5}{12}; b_0 = \frac{8}{12}; b_1 = -\frac{1}{12}$ $x_{n+1} = x_n + \frac{h}{12}[5f_{n+1} + 8f_n - f_{n-1}]$	$\frac{1}{24}h^4 x^{(4)}(\xi)$

12.4.4. Tính chính xác và tính ổn định của phương pháp tích phân số

Tính chính xác

Lỗi cắt tại bước thời gian tỉ lệ nghịch với đạo hàm bậc cao hơn và kích thước bước h. Nếu có thể *ước tính* lỗi cắt cục bộ tại mỗi bước, thì có thể điều chỉnh tăng hoặc giảm kích thước bước h. Tổng quát, các phương pháp ẩn bậc cao hơn là tốt hơn vì chúng tạo ra lỗi nhỏ hơn, tuy nhiên phải giải phương trình phi tuyến tại mỗi bước. Giảm kích thước bước sẽ làm giảm lỗi cắt cục bộ đồng thời cũng tăng tốc độ hội tụ của nghiệm lặp tại mỗi bước. Giảm bước thời gian, sẽ tăng khối lượng tính toán.

Lỗi cục bộ tích lũy thành lỗi tổng. Trong khi có thể ước tính được lỗi cắt cục bộ, nhưng đáng tiếc, không có sẵn các thủ tục chung để khống chế sự tăng lỗi tổng. Vì vậy, hầu hết các phương pháp dựa vào ước tính lỗi cắt cục bộ và giảm kích cỡ bước khi cần thiết.

Tính ổn định

Tính ổn định đánh giá mức độ nghiệm đệ qui hội tụ đến nghiệm đúng. Tính ổn định phụ thuộc vào bài toán cụ thể và phương pháp tích phân, thực tế thường nghiên cứu tính ổn định của các phương pháp tích phân bằng cách dùng một bài toán kiểm tra đơn giản như:

$$\dot{x} = \lambda x$$

Trong đó nghiệm được biết là:

$$x(t) = e^{\lambda t}$$

Nếu $\mathrm{Re}(\lambda) < 0$, thì nghiệm giá trị thực xu hướng tiến đến 0 khi $t \to \infty$. Với trường hợp kiểm tra này, tích phân được nói là ổn định nếu nghiệm đệ quy hội tụ tới 0 khi $n \to \infty$.

Trong khi dễ dàng nghiên cứu tính ổn định của các phương pháp tích phân đối với trường hợp kiểm tra đơn giản, thì khó mà đưa ra các kết luận chung về tính ổn định của một phương pháp khi ứng dụng vào một phương trình vi phân phi tuyến tùy ý. Thông thường phương pháp ẩn có thuộc tính ổn định tốt hơn phương pháp hiện.

Thật khó để bám lỗi cắt và tính ổn định của nghiệm NLDE, vì việc bám lỗi cắt đòi hỏi phải biết các đạo hàm bậc cao hơn, và việc phân tích tính ổn định cần phải biết nghiệm phương trình đặc trưng của phiên bản NLDE đã được tuyến tính hóa tại lân cận t_n. Trong quá trình mô phỏng, một phương pháp khám phá được dùng gồm việc so sánh các nghiệm để có được các bước thời gian h và $h/2$. Nếu lỗi ước tính liên quan với cả hai bước thời gian này nằm trong các giới hạn cụ thể, thì nghiệm được chấp nhận. Nếu không, phải xác định lại bước thời gian và quá trình mô phỏng được lặp lại. Phương pháp này kiểm soát được lỗi cắt và tính ổn định.

Giữa các kỹ thuật tích phân sẵn có, không thể nói cái nào là "tốt nhất" vì lời giải phụ thuộc vào bài toán được xét, tính chính xác, kiểu đầu ra, và khối lượng tính toán. Luật hình thang, phương pháp R-K bậc 4, phương pháp Adam-Moulton (với dự đoán-hiệu chỉnh) là những phương pháp được sử dụng phổ biến nhất tạo ra sự dung hòa giữa độ phức tạp tính toán, tính chính xác và tính ổn định (phương pháp hình thang ít phức tạp nhất). Vì các phần tử phi tuyến biểu lộ giống như PLL, nên một trong các phương pháp này có kích thước bước nhỏ (bội số 8mẫu/1 Hz của độ rộng băng tần) cho ta nghiệm khá chính xác và ổn định.

12.4.5. Nghiệm của NLDE bậc cao hơn - Trường hợp Vectơ

Nếu mô hình phương trình vi phân phi tuyến NLDE bậc m, thì có thể chuyển **nó thành m** phương trình vi phân bậc một đồng thời (như được mô tả phần 12.4.1), ở dạng vecto là:

$$\dot{\mathbf{X}} = F(\mathbf{X}, \mathrm{t}, \mathbf{U})$$

Trong đó, X, U là các vecto kích cỡ $mx1$. Nghiệm lặp đa bước của NLDE vectơ có cùng dạng như trường hợp vô hướng và được cho bởi:

$$X_{n+1} = \sum_{i=0}^{p} a_i X_{n-i} + \sum_{j=-1}^{p} b_j \dot{X}_{n-j} = \sum_{i=0}^{p} a_i X_{n-i} + \sum_{j=-1}^{p} b_j F_{n-j}$$

Trong đó a_i, b_j là các hằng số vô hướng được cho trong bảng 12.1.

Với phương pháp ẩn, tại mỗi bước phải giải một tập các phương trình phi tuyến *đồng thời*. Khi này, có thể dùng phương pháp dự đoán-hiệu chỉnh hoặc phương pháp N-R. Dạng vecto của phương pháp N-R được cho bởi:

$$G(X_{n+1}) = X_{n+1} - \sum_{i=0}^{p} a_i X_{n-i} + \sum_{j=-1}^{p} b_j F_{n-j} \quad (12.111)$$

$$X_{n+1}^{r+1} = X_{n+1}^{r} - J^{-1}(X_{n+1}^{r}).G(X_{n+1}^{r}), \quad r = 1,2,3,\ldots \quad (12.112)$$

Trong đó, thực thể thứ (i,j) trong ma trận Jacobian $m \times m$, J(.) được định nghĩa là:

$$\left[J(X_{n+1}^{r})\right]_{i,j} = \frac{\partial g_i(X_{n+1}^{r})}{\partial x_j} \quad (12.113)$$

Tìm giá trị bắt đầu X^{I}_{n+1} của X_{n+1} bằng phương pháp hiện. Tại mỗi bước thời gian t_{n+1}, thực hiện lặp (chỉ số lặp r) cho đến khi sự khác nhau các nghiệm lặp là nhỏ. Sau đó, tăng chỉ số thời gian lên t_{n+2}, tìm giá trị khởi đầu X^{I}_{n+2} của X_{n+2} bằng phương pháp hiện, sau đó tìm nghiệm lặp cho X_{n+2}. Quá trình lặp lại cho đến khi chỉ số thời gian tiến đến thời điểm dừng mô phỏng.

12.5. Mô phỏng PLL minh hoạ thiết bị phi tuyến

Tại đây, ta trở lại bài toán PLL trong chương 6 và minh họa việc mô phỏng PLL sử dụng một vài kỹ thuật tích phân đã được trình bày ở trên. Dùng các kỹ thuật được định xác định ở dạng bài toán PLL trong các phần dưới đây.

Phương pháp Euler tiến (phương pháp hiện)

Xác định các phương trình như sau:

$$t_{n+1} = t_n + h \quad (12.114)$$

$$x_{1,n+1} = x_{1,n} + hx_{2,n} \quad (12.115)$$

$$x_{2,n+1} = x_{2,n} + h\left[\sin(\phi_n - c_1 x_{2,n} - c_2 x_{1,n}) + c_3 x_{2,n}\right] \quad (12.116)$$

Phương pháp Euler lùi (phương pháp ẩn với dự đoán-hiệu chỉnh)

Bộ dự đoán được định nghĩa bởi:

$$x_{1,n+1}^{1} = x_{1,n} + hx_{2,n} \quad (12.117)$$

$$x^1_{2,n+1} = x_{2,n} + h[\sin(\phi_n - c_1 x_{2,n} - c_2 x_{1,n}) + c_3 x_{2,n}] \tag{12.118}$$

Bộ hiệu chỉnh được định nghĩa bởi:

$$x^{r+1}_{1,n+1} = x^r_{1,n} + hx^r_{2,r}, \quad r = 1,2,\ldots \tag{12.119}$$

$$x^{r+1}_{2,n+1} = x_{2,n} + h\left[\sin\left(\phi_{n+1} - c_1 x^r_{2,n+1} - c_2 x^r_{1,n+1}\right) + c_3 x^r_{2,n+1}\right], \quad r = 1,2,\ldots \tag{12.120}$$

Phương pháp Euler lùi (phương pháp ẩn với các phép lặp N-R)

Bộ dự đoán giống với phương pháp Euler lùi [xem (12.117) và (12.118)]. Các phép lặp N-R cho X_{n+1} được xác định bởi:

$$X^{r+1}_{n+1} = X^r_{n+1} - J^{-1}(X^r_{n+1}).G(X^r_{n+1}) \tag{12.121}$$

Trong đó :

$$G(X^r_{n+1}) = \begin{bmatrix} g_1(X^r_{n+1}) \\ g_2(X_{n+1}) \end{bmatrix} = \begin{bmatrix} x^r_{1,n+1} - x^r_{1,n} - hx^r_{2,n+1} \\ x^r_{2,n+1} - x^r_{2,n} + h[\sin(\phi_n - c_1 x^r_{2,n} - c_2 x^r_{1,n}) + c_3 x^r_{2,n}] \end{bmatrix} \tag{12.122}$$

Và $J^{-1}(X^r_{n+1})$ là ma trận 2x2 :

$$J^{-1}(X^r_{n+1}) = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} \tag{12.123}$$

Với các phần tử :

$$e_{11} = 1 \tag{12.124}$$

$$e_{12} = -h \tag{12.125}$$

$$e_{21} = -c_2 h[\cos(\phi_{n+1} - c_1 x^r_{2,n+1} - c_2 x^r_{1,n+1}) + c_3 x^r_{2,n+1}] \tag{12.126}$$

$$e_{22} = 1 - h[c_1 \cos(\phi_{n+1} - c_1 x^r_{2,n+1} - c_2 x^r_{1,n+1}) + c_3] \tag{12.127}$$

Chương 13
MÔ HÌNH HÓA VÀ MÔ PHỎNG HỆ THỐNG THAY ĐỔI THEO THỜI GIAN

13.1 Mở đầu

Tất cả các phần tử trong hệ thống và mô hình đã được nghiên cứu đều cố định hoặc không thay đổi theo thời gian. Trong các chương trước, ta đã thảo luận chi tiết về việc mô hình hóa và mô phỏng các mô hình tuyến tính bất biến LTIV. Chương này ta sẽ nghiên cứu các phần tử và hệ thống truyền thông ở mức cao hơn là: mô hình hóa và mô phỏng các hệ thống thay đổi theo thời gian LTV. Ta xét nhiều ví dụ để minh họa tính phụ thuộc vào thời gian của hệ thống, điển hình nhất là kênh thay đổi theo thời gian trong hệ thống thông tin di động.

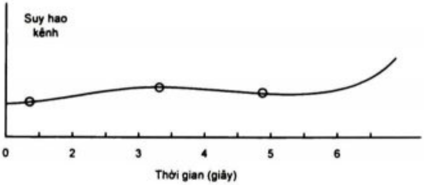
Giả định tính bất biến theo thời gian được hiểu là các tính chất của hệ thống không thay đổi theo thời gian. Nếu ta mô hình hóa hệ thống bất biến theo thời gian bằng một hàm truyền đạt, thì tính bất biến theo thời gian xác định hàm truyền đạt (cả biên độ và pha) sẽ không đổi, nó là một hàm của thời gian. Chẳng hạn: xét mô hình bộ lọc Butterworth bậc 3 có độ rộng băng thông 3 dB là 1MHz. Tính bất biến theo thời gian nghĩa là cả bậc và độ rộng băng thông của bộ lọc đều không thay đổi, là một hàm thời gian. Bộ lọc chỉ không thay đổi theo thời gian nếu các giá trị của tất cả các phần tử vật lý, như điện trở, tụ điện, và các tham số của các phần tử tích cực như hệ số khuếch đại của bộ khuếch đại, đều không thay đổi theo thời gian. Trong khi giả định bất biến theo thời gian có thể được duy trì trong một khoảng thời gian ngắn, thì khi xét trong các khoảng thời gian dài do sự lão hóa làm cho chúng trở nên thay đổi theo thời gian. Từ quan điểm thực tế, dù hệ thống được xét có bất biến theo thời gian hay không, nó không chỉ phụ thuộc vào hệ thống mà còn phụ thuộc vào bản chất của bài toán được giải quyết.

Dù có sử dụng mô hình hệ thống bất biến theo thời gian hay phụ thuộc vào thời gian đi chăng nữa thì chúng thường được xác định bởi tốc độ mà tại đó các đặc tính của hệ thống sẽ được mô hình hóa là thay đổi so với các thông số khác của hệ thống (chẳng hạn tốc độ ký hiệu). Ví dụ: nếu hằng số thời gian liên quan đến tính thay đổi thời gian là rất lớn so với tốc độ ký hiệu thì mô hình bất biến theo thời gian được hiệu chỉnh. Nói cách khác, nếu các tham số hệ thống thay đổi tại tốc độ gần với tốc độ ký hiệu thì mô hình thay đổi theo thời gian là hợp lệ. Vì vậy, dẫn đến khái niệm thay đổi "*nhanh*" hay "*chậm*" so với tốc độ ký hiệu hoặc một số thuộc tính khác của hệ thống ảnh hưởng đến việc lựa chọn mô hình thay đổi theo thời gian hay không thay đổi theo thời gian. Để được tường minh, ta khảo sát bài toán này qua hai trường hợp điển hình.

13.1.1. Hệ thống thay đổi theo thời gian điển hình

Trường hợp 1: Xét hệ thống truyền thông vô tuyến viba trong đó các ăng ten phát và thu được đặt trên các tháp viba cố định, "kênh" giữa ăng ten phát thu là không khí và việc thay đổi các đặc tính kênh là do điều kiện không khí thay đổi, điển hình là có các hằng số thời gian phút

hoặc giờ. Nếu tuyến thông tin đang hoạt động tại tốc độ ký hiệu 100 Mbit/s thì hằng số thời gian liên quan đến thay đổi của kênh là rất dài so với thời gian ký hiệu 10^{-8}s. Khi này, kênh sẽ gần như giữ nguyên trạng thái trong khi hàng triệu ký hiệu truyền qua tuyến. Nếu mục đích của mô phỏng là ước tính BER, thì kênh có thể được giả định là ở trạng thái "*tĩnh*" và có thể sử dụng mô hình bất biến theo thời gian trong mỗi lần ước tính BER. Tất nhiên, kết quả ước tính BER chỉ có giá trị duy nhất đối với trạng thái kênh cụ thể. Tính cách của kênh trong thời gian dài, và ảnh hưởng của nó lên hiệu năng hệ thống thời gian dài, có thể được ước lượng bằng cách phân tích hiệu năng hệ thống trên một chuỗi các "*chớp ảnh*" của các điều kiện kênh tĩnh, sử dụng một mô hình bất biến theo thời gian khác cho mỗi chớp ảnh. Điều này được minh họa trong hình 13.1, thể hiện 3 mô phỏng được triển khai cho 3 giá trị suy hao kênh. Từ những mô phỏng như vậy một khi có được những phép đo hiệu năng như "*xác suất ngừng hoạt động*", mô tả phần thời gian trong đó hiệu năng kênh có thể giảm dưới một số ngưỡng BER. Nếu mô phỏng 10000 chớp ảnh của kênh và 100 trong số các điều kiện kênh này tạo ra BER ứng với hiệu năng hệ thống không thỏa mãn, ví dụ BER > 10^{-3} thì xác suất ngừng hoạt động là (100/10000) = 0,01 đối với ngưỡng BER cụ thể này.



Hình 13.1: Các chớp ảnh của kênh thay đổi chậm theo thời gian

Trường hợp 2: Xét một hệ thống thông tin di động gồm một trạm gốc cố định và một người dùng di động. Các đặc tính của kênh truyền thông giữa máy phát và máy thu sẽ thay đổi theo thời gian, vì các tham số của kênh như suy hao và trễ là thay đổi theo thời gian do sự chuyển động tương đối giữa trạm gốc và người dùng di động. Ngoài ra, sự thay đổi trong các điều kiện không khí cũng góp phần vào bản chất thay đổi theo thời gian của kênh. Nếu người dùng di động chuyển động nhanh và nếu tốc độ ký hiệu khoảng 10.000 ký hiệu/s, thì tốc độ (tại đó các điều kiện kênh thay đổi) có thể so sánh được với tốc độ ký hiệu. Trong trường hợp này dùng mô hình kênh thay đổi theo thời gian.

13.1.2. Mô hình hóa và giải pháp mô phỏng

Cũng như hệ thống tuyến tính bất biến LTIV, hệ thống tuyến tính thay đổi theo thời gian LTV cũng được mô hình hóa và mô phỏng trong cả miền thời gian và miền tần số. Giải pháp trong miền thời gian dẫn đến mô hình phải chứa cấu trúc đường trễ rẽ nhánh có các độ lợi nhánh thay đổi theo thời gian. Mô hình này rất dễ để thực hiện cho các mục đích mô phỏng và tính toán hiệu quả nếu đáp ứng xung kim thay đổi theo thời gian là tương đối ngắn.

Nhiều khái niệm mô hình hóa và mô phỏng được đề cập ở các chương trước cho các hệ thống tuyến tính bất biến LTIV được áp dụng cho các hệ thống tuyến tính thay đổi theo thời gian LTV, nhưng có vài điểm khác biệt quan trọng. Cần phải đặc biệt lưu ý tốc độ lấy mẫu bởi

lẽ cần phải tăng tốc độ lấy mẫu do sự nở rộng băng thông từ tính cách thay đổi theo thời gian. Nguồn gốc của sự nở rộng băng thông là do trải rộng "*doppler*" trong hệ thống thông tin di động. Ngoài ra, phải thận trọng khi đơn giản hóa sơ đồ khối hệ thống LTV, vì các khối LTV không tuân theo *tính chất giao hoán* do đó thứ tự thực hiện không thể đổi giữa các khối LTV với nhau được như khi xét cho các khối LTIV.

Tuy nhiên, nếu hệ thống thay đổi theo thời gian có tính tuyến tính, thì áp dụng xếp chồng, tích chập và có thể sử dụng một số kỹ thuật phân tích trong miền thời gian và miền tần số đã được triển khai cho hệ thống LTIV vào việc mô hình hóa và mô phỏng hệ thống LTV với những sửa đổi không đáng kể. Ta cũng sẽ dùng phương pháp biểu diễn tín hiệu và hệ thống thông thấp tương đương khi triển khai các mô hình mô phỏng và kỹ thuật mô phỏng cho các hệ thống thay đổi theo thời gian.

13.2. Các mô hình cho các hệ thống tuyến tính thay đổi theo thời gian LTV

Trong miền thời gian, hệ thống LTIV được mô tả bởi một đáp ứng xung kim đường bao phức $\tilde{h}(\tau)$, $\tilde{h}(\tau)$ được định nghĩa là đáp ứng của hệ thống ở thời điểm τ đối với một xung kim được đặt vào hệ thống ở thời điểm $t = 0$. Biến τ thể hiện " *thời gian trôi qua*", là hiệu số thời gian giữa thời điểm đo được đáp ứng xung kim và thời điểm đưa xung kim vào hệ thống. Quan hệ vào/ra đường bao phức đối với hệ thống LTIV được cho bởi tích chập quen thuộc.

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{h}(\tau)\tilde{x}(t-\tau)d\tau \tag{13.1}$$

Trong đó $\tilde{x}(t)$ và $\tilde{y}(t)$ trình bày các đường bao phức vào/ra của hệ thống.

Biến đổi Fourier ngược của (13.1) tạo ra quan hệ vào/ra hệ thống LTIV trong miền tần số.

$$\tilde{Y}(f) = \tilde{X}(f)\tilde{H}(f) \tag{13.2}$$

Trong đó $\tilde{H}(f)$ là hàm truyền đạt của hệ thống, và $\tilde{X}(f)$ và $\tilde{Y}(f)$ là biến đổi Fourier của đầu vào/ ra tương ứng. Tìm đầu ra $\tilde{y}(t)$ bằng cách biến đổi ngược của $\tilde{Y}(f)$.

$$\begin{aligned}\tilde{y}(t) &= \int_{-\infty}^{\infty} \tilde{Y}(f)e^{j2\pi ft}\,df \\ &= \int_{-\infty}^{\infty} \tilde{X}(f)\tilde{H}(f)e^{j2\pi ft}\,df\end{aligned} \tag{13.3}$$

Các biểu thức trên được dùng như điểm bắt đầu để rút ra các mô hình cho các hệ thống thay đổi thời gian.

13.2.1. Mô tả hệ thống LTV trong miền thời gian

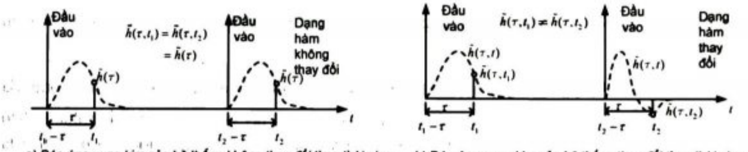
Các hệ thống thay đổi theo thời gian cũng được đặc điểm hóa trong miền thời gian bằng một đáp ứng xung kim.

Với hệ thống thay đổi theo thời gian, đáp ứng xung kim có dạng $\tilde{h}(\tau,t)$ được định nghĩa là đáp ứng của hệ thống được đo ở thời điểm t đối với một xung kim được đặt vào hệ thống ở

thời điểm sớm hơn τ giây. Nói một cách khác, xung kim được đưa vào hệ thống tại thời điểm $t-\tau$ và đo được đáp ứng của hệ thống tại thời điểm t, sau một "*thời gian trôi*" là τ. Do tính thay đổi theo thời gian, nên đáp ứng xung kim sẽ thay đổi là một hàm hai biến $(t-\tau$ và $t)$ trong đó $t-\tau$ là thời điểm xung kim được đặt vào hệ thống và t là thời điểm đo được ở đầu ra của hệ thống.

Với hệ thống bất biến theo thời gian, đáp ứng xung chỉ là một hàm của thời gian trôi τ được ký hiệu là $\tilde{h}(\tau)$.

Lưu ý rằng, đáp ứng xung của hệ thống thay đổi theo thời gian là một hàm của 2 đối số, trong khi đó đáp ứng xung của hệ thống bất biến theo thời gian chỉ có một đối số. Hình 13.2 phác hoạ đáp ứng xung của hệ thống thay đổi và không thay đổi theo thời gian.



a) Đáp ứng xung kim của hệ thống không thay đổi theo thời gian *b) Đáp ứng xung kim của hệ thống thay đổi theo thời gian*

Hình 13.2: Đáp ứng xung kim của hệ thống thay đổi và không thay đổi theo thời gian

Trong khi đáp ứng xung của hệ thống LTIV duy trì cùng *dạng hàm* không phụ thuộc vào thời điểm đưa xung kim vào đầu vào, thì đáp ứng xung của hệ thống LTV lại phụ thuộc vào thời điểm đưa xung kim hệ thống. Nói cách khác,

Với hệ thống LTIV, thì:

$$\tilde{h}(\tau,t_1)=\tilde{h}(\tau,t_2)=\tilde{h}(\tau) \quad (13.4)$$

Với hệ thống LTV, thì:

$$\tilde{h}(\tau,t_1)\neq\tilde{h}(\tau,t_2) \quad (13.5)$$

Ngoài ra, đáp ứng của hệ thống LTIV đối với một đầu vào tùy ý vẫn duy trì giống nhau bất chấp thời điểm đưa tín hiệu vào hệ thống ngoại trừ một trễ thời gian. Nếu ở thời điểm t nào đó, ta đưa tín hiệu $\tilde{x}(t)$ vào hệ thống bất biến nhận được đầu ra là $\tilde{y}(t)$, thì cùng đầu vào đó được đưa vào hệ thống ở thời điểm chậm hơn t_0 là $\tilde{x}(t-t_0)$, ta sẽ nhận được ở đầu ra của hệ thống đó một phiên bản bị trễ $\tilde{y}(t-t_0)$ của $\tilde{y}(t)$. Trong hệ thống thay đổi theo thời gian, các đáp ứng của hệ thống đối với các đầu vào giống nhau có thể hoàn toàn khác nhau nếu các đầu vào giống nhau đó được đặt vào hệ thống tại các thời điểm khác nhau.

Cần lưu ý rằng, sự khác biệt quan trọng giữa đáp ứng xung kim của hệ thống LTIV & LTV là: trong khi đáp ứng xung kim của hệ thông LTIV chỉ là một hàm của thời gian trôi τ và không phụ thuộc vào thời điểm đưa tín hiệu vào hệ thống, thì đáp ứng xung kim của hệ thống LTV là một hàm hai biến (thời gian trôi τ và thời điểm quan sát t).

Hai biến thời gian τ và t trong $\tilde{h}(\tau,t)$ đặc điểm hóa cho hai khía cạnh khác nhau của hệ thống. Biến τ có vai trò giống như biến τ trong đáp ứng xung kim $\tilde{h}(\tau)$ của hệ thống bất biến

theo thời gian do đó biến đổi Fourier theo biến τ dẫn đến khái niệm hàm truyền đạt, đáp ứng tần số, độ rộng băng tần. Đối với hệ thống LTV, một khi triển khai khái niệm hàm truyền đạt bằng cách biến đổi Fourier hàm đáp ứng xung kim $\tilde{h}(\tau,t)$ theo biến τ nhận được hàm truyền đạt $\tilde{H}(f,t)$, là một hàm truyền đạt biến đổi theo thời gian:

$$\tilde{H}(f,t)=\int_{-\infty}^{\infty}\tilde{h}(\tau,t)e^{-j2\pi f\tau}\,d\tau \tag{13.6}$$

Nếu hệ thống "*thay đổi chậm theo thời gian*", thì khái niệm đáp ứng tần số và độ rộng băng tần có thể được áp dụng cho $\tilde{H}(f,t)$. Trong khi hệ thống LTIV được đặc trưng hóa bởi *một* hàm đáp ứng xung kim và *một* hàm truyền đạt, thì hệ thống LTV được đặc trưng hóa bởi *một họ* các hàm đáp ứng xung và một họ các hàm truyền đạt, với mỗi hàm ứng với mỗi giá trị của *t*. Nếu $\tilde{h}(\tau,t)=\tilde{h}(\tau)$, hoặc tương ứng $\tilde{H}(f,t)=\tilde{H}(f)$, thì hệ thống là hệ thống bất biến theo thời gian.

Biến *t* trong $\tilde{h}(\tau,t)$ và $\tilde{H}(f,t)$ mô tả bản chất thay đổi theo thời gian của hệ thống. Sự phụ thuộc mạnh mẽ vào *t*, và sự thay đổi nhanh theo *t*, cho thấy hệ thống thay đổi nhanh theo thời gian. Thông thường, bản chất thay đổi theo thời gian của hệ thống được mô hình hóa là một hiện tượng ngẫu nhiên, và $\tilde{h}(\tau,t)$ được xử lý như một quá trình ngẫu nhiên theo *t*. Nếu quá trình là dừng, thì các thay đổi theo thời gian có thể được mô hình hóa bởi một hàm tự tương quan trong miền thời gian hoặc bởi một hàm mật độ phổ công suất PSD trong miền tần số. Hằng số thời gian của hàm tự tương quan hoặc độ rộng băng tần của hàm mật độ phổ công suất là các *thông số then chốt* mô tả hệ thống có đáp ứng xung kim $\tilde{h}(\tau,t)$ là thay đổi *nhanh* hay *chậm* theo thời gian.

Từ định nghĩa đáp ứng xung kim của hệ thống LTV, dễ dàng thấy rằng quan hệ vào/ra được biểu diễn bởi tích chập sau:

$$\tilde{y}(t)=\int_{-\infty}^{\infty}\tilde{h}(\tau,t)\tilde{x}(t-\tau)d\tau=\int_{-\infty}^{\infty}\tilde{h}(t-\tau)\tilde{x}(\tau)d\tau \tag{13.7}$$

Phiên bản miền tần số của quan hệ vào/ra có phần phức tạp hơn như sẽ thấy ở phần sau.

13.2.2. Mô tả hệ thống LTV trong miền tần số

Như một điểm bắt đầu để mô tả hệ thống LTV trong miền tần số, ta định nghĩa biến đổi Fourier hai chiều của $\tilde{h}(\tau,t)$ là:

$$\tilde{H}(f_1,f_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\tilde{h}(\tau,t)e^{-j2\pi(f_1\tau+f_2t)}\,d\tau dt \tag{13.8}$$

Trong quá trình định nghĩa biến đổi Fourier hai chiều, thường giả định hàm đáp ứng xung kim có năng lượng hữu hạn nhằm đảm bảo sự tồn tại phép biến đổi Fourier. Tuy nhiên ta sẽ thấy rằng, để mô hình hóa kênh, thì $\tilde{h}(\tau,t)$ sẽ được coi là một quá trình ngẫu nhiên dừng theo *t* trong đó có thể không tồn tại biến đổi Fourier của $\tilde{h}(\tau,t)$ theo t. Tất nhiên, hàm tự tương

quan sẽ tồn tại và thủ tục phù hợp để định nghĩa hàm tự tương quan của $\tilde{h}(\tau,t)$ theo biến t, và sau đó lấy biến đổi Fourier của hàm tự tương quan nhận được biểu diễn trong miền tần số. Kết quả ta có hàm mật độ phổ công suất của quá trình ngẫu nhiên đó. Cần phải lưu ý rằng, nếu hệ thống là bất biến theo thời gian, thì $\tilde{h}(\tau,t)=\tilde{h}(\tau)$ và $\tilde{H}(f_1,f_2)=\tilde{H}(f_1).\delta(f_2)$.

Lấy biến đổi Fourier ngược của $\tilde{H}(f_1,f_2)$ được:

$$\tilde{h}(\tau,t)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\tilde{H}(f_1,f_2)e^{j2\pi(f_1\tau+f_2t)}\,df_1df_2 \quad (13.9)$$

Thay $\tilde{h}(\tau,t)$ vào (13.7) ta có:

$$\tilde{Y}(f)=\int_{-\infty}^{\infty}\tilde{H}\left(f_1,f-f_1\right).\tilde{X}\left(f_1\right)df_1 \quad (13.10)$$

$$\tilde{y}(t)=\int_{-\infty}^{\infty}\tilde{Y}(f)e^{j2\pi ft}\,df \quad (13.11)$$

Trong "*hàm truyền đạt*" hai chiều được định nghĩa ở (13.8), thì biến tần số f_1 liên quan với biến thời gian τ và được xét tương tự như biến tần số trong hàm truyền đạt $H(f)$ của các hệ thống LTIV. Tuy nhiên, quan hệ vào/ra của hệ thống LTV trong miền tần số được cho ở (13.10) chứa một phép tích chập trong miền tần số đối với biến thứ 2 của hàm truyền đạt $\tilde{H}(f_1,f_2)$. Phép tích chập này giải thích ảnh hưởng của bản chất thay đổi theo thời gian của hệ thống trong miền tần số.

Hệ thống LTIV

Nếu đưa vào hệ thống LTIV là một đơn mang tại f_c+f_0

$$x(t)=A\cos\left[2\pi\left(f_c+f_0\right)t\right] \quad (13.12)$$

có đường bao phức là:

$$\tilde{x}(t)=Ae^{j2\pi f_0t} \quad (13.13)$$

thì quan hệ vào/ra trong miền tần số được cho bởi:

$$\tilde{X}(f)=A.\delta\left(f-f_0\right) \quad (13.14)$$

$$\tilde{Y}(f)=\tilde{H}(f).\tilde{X}(f)=\tilde{H}(f_0).A.\delta\left(f-f_0\right) \quad (13.15)$$

Đầu ra đường bao phức hệ thống trong miền thời gian được định nghĩa là:

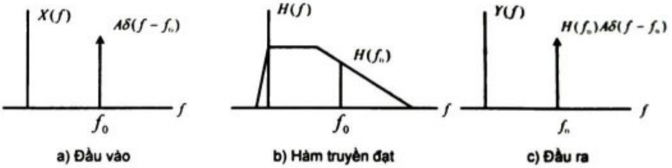
$$\tilde{y}(t)=A\left|\tilde{H}(f_0)\right|e^{\left[j2\pi f_0t+\angle\tilde{H}(f_0)\right]} \quad (13.16)$$

Và tín hiệu thông dải trong miền thời gian được cho bởi:

$$y(t)=A\left|\tilde{H}(f_0)\right|\cos\left[2\pi(f_c+f_0)t+\angle\tilde{H}(f_0)\right] \quad (13.17)$$

Quan hệ trong (13.13) và (13.16) đối với các hệ thống LTIV cho thấy rằng, khi đưa vào hệ thống một đơn mang phức tại tần số f_0, thì hệ thống tạo ra một đơn mang đầu ra cũng tại tần

số f_0 đó. Biên độ và pha của đơn mang đầu ra bị ảnh hưởng bởi đáp ứng biên độ và đáp ứng pha của hệ thống cũng tại tần số f_0. Điều này được minh hoạ ở hình 13.3.



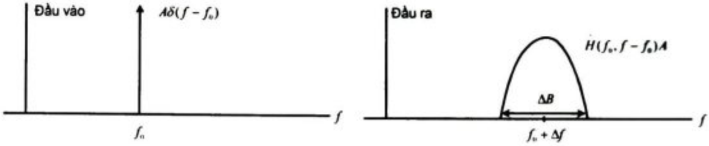
Hình 13.3: Đáp ứng của hệ thống LTIV đối với một đầu vào đơn mang

Hệ thống LTV

Ta xét cùng một trạng thái đối với hệ thống thay đổi theo thời gian. Với cùng đầu vào (13.13), thì đầu ra của hệ thống LTV trong miền tần số là:

$$\tilde{Y}(f) = \int_{-\infty}^{\infty} \tilde{H}(f_1, f - f_1).\tilde{X}(f_1).df_1 = \int_{-\infty}^{\infty} \tilde{H}(f_1, f - f_1).A.\delta(f_1 - f_0).df_1 = A.\tilde{H}(f_0, f - f_0) \qquad (13.18)$$

Phương trình trên cho thấy, đầu ra không chứa một đơn mang đơn mà có thể gồm một ***dải tần số***. Một minh họa được cho trong hình 13.4. *Lưu ý rằng*, nếu hệ thống là bất biến theo thời gian, thì $\tilde{h}(\tau,t) = \tilde{h}(\tau)$, $\tilde{H}(f_1, f_2) = \tilde{H}(f_1).\delta(f_2)$ và (13.18) tạo ra $\tilde{Y}(f) = \tilde{H}(f_0).A.\delta(f - f_0)$ giống như (13.15).



Hình 13.4: Đáp ứng của hệ thống LTV đối với một đầu vào đơn mang

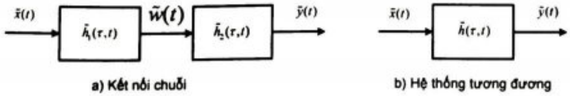
Tổng quát, đầu ra của hệ thống LTV tương ứng với một đơn mang đầu vào tại tần số f_0 có thể bị dịch tần số và cũng có thể bị trải rộng. Trong kịch bản kênh truyền thông di động, thì điều này được coi là *dịch Doppler* và *trải rộng Doppler* và được tạo ra bởi sự chuyển động tương đối giữa ăng ten phát và ăng ten thu, hoặc bởi những sự thay đổi khác trong kênh.

13.2.3. Các tính chất của hệ thống LTV

Một vài tính chất của hệ thống LTIV là hữu hiệu để đơn giản hoá mô hình mô phỏng các hệ thống truyền thông. Ví dụ việc kết hợp các hàm truyền đạt của các khối mắc song song, các khối mắc nối tiếp, kết hợp các khối nối tiếp và song song. Những phép toán này dựa trên các tính chất kết hợp, phân phối và giao hoán của các hệ thống LTIV. Những tính chất này được kiểm tra cho các hệ thống LTV.

Tính kết hợp

Để kiểm tra tính chất kết hợp một cách chi tiết, xét hệ thống được cho ở hình 13.5(a).



Hình 13.5: Kết nối nối tiếp và hệ thống tương đương

Đầu ra của khối đầu tiên trong hình 15.5(a) được cho bởi:

$$\tilde{w}(t) = \int_{-\infty}^{\infty} \tilde{h}_1(\tau_1,t).\tilde{x}(t-\tau_1)d\tau_1 = \int_{-\infty}^{\infty} \tilde{h}_1(t-\tau_1,t).\tilde{x}(\tau_1)d\tau_1 \tag{13.19}$$

Đầu ra của khối thứ 2 là:

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{h}_2(\tau_2,t).\tilde{w}\left(t-\tau_2\right)d\tau_2 \tag{13.20}$$

Thay $\tilde{w}(t)$ vào (13.20) ta được:

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{h}_2(\tau_2,t)\left[\int_{-\infty}^{\infty} \tilde{h}_1(t-\tau_2-\tau_1,t-\tau_2).\tilde{x}(\tau_1)d\tau_1\right]d\tau_2 \tag{13.21}$$

Có thể được viết là:

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{x}(\tau_1)\underbrace{\left[\int_{-\infty}^{\infty} \tilde{h}_1(t-\tau_2-\tau_1,t-\tau_2)\tilde{h}_2(\tau_2,t)d\tau_2\right]}_{\tilde{h}(t-\tau_1,t)\text{: Đáp ứng xung kim toàn bộ của hệ thống}}d\tau_1 \tag{13.22}$$

Đại lượng bên trong ngoặc vuông là đáp ứng xung kim toàn bộ của hệ thống $\tilde{h}(t-\tau_1,t)$.

Thay $\tau = t-\tau_1$, ta được đáp ứng xung kim chung của 2 khối nối tầng là:

$$\tilde{h}(\tau,t) = \int_{-\infty}^{\infty} \tilde{h}_1(\tau-\tau_2,t-\tau_2).\tilde{h}_2(\tau_2,t)d\tau_2 \tag{13.23}$$

Ở dạng đơn giản hơn ta có:

$$\tilde{h}(\tau,t) = \int_{-\infty}^{\infty} \tilde{h}_1(\tau-\alpha,t-\alpha)\tilde{h}_2(\alpha,t)d\alpha \tag{13.24}$$

Sử dụng phương trình trước có thể định nghĩa tính chất kết hợp cho hệ thống LTV là:

$$\tilde{x}(\tau)\otimes\tilde{h}_1(\tau)\otimes\tilde{h}_2(\tau) = \left[\tilde{x}(\tau)\otimes\tilde{h}_1(\tau)\right]\otimes\tilde{h}_2(\tau) = \tilde{x}(\tau)\otimes\left[\tilde{h}_1(\tau)\otimes\tilde{h}_2(\tau)\right] \tag{13.25}$$

Đáp ứng xung kim toàn bộ là:

$$\tilde{h}(\tau,t) = \tilde{h}_1(\tau,t)\otimes\tilde{h}_2(\tau,t) \tag{13.26}$$

$$\tilde{h}(\tau,t) = \int_{-\infty}^{\infty} \tilde{h}_1(\tau-\alpha,t-\alpha)\tilde{h}_2(\alpha,t)d\alpha \tag{13.27}$$

Như được thấy ở hình 13.5(b).

Tính giao hoán

Trong khi có thể kết hợp đáp ứng của hai khối LTV nối tiếp theo (13.27) thì không thể thay đổi thứ tự các khối LTV cho nhau được. Việc đổi chỗ hai khối cho nhau tạo ra đáp ứng xung kim tổng thể là tích chập của các đáp ứng xung kim riêng biệt. Vì vậy,

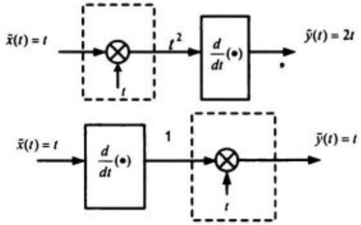
$$\tilde{g}(\tau,t) = \tilde{h}_2(\tau,t) \otimes \tilde{h}_1(\tau,t) = \int_{-\infty}^{\infty} \tilde{h}_2(\tau-\alpha,t-\alpha)\tilde{h}_1(\alpha,t)d\alpha \tag{13.28}$$

Tổng quát, $\tilde{h}(\tau,t) \neq \tilde{g}(\tau,t)$. Vì vậy ở dạng tổng quát, việc thay đổi thứ tự của LTV không phải là phép toán hợp lệ. Ví dụ được cho ở hình 13.6, cho ta một minh chứng về tính chất giao hoán không dùng được đối với các khối LTV, nghĩa là với các hệ thống LTV thì:

$$\tilde{h}_1(\tau,t) \otimes \tilde{h}_2(\tau,t) \neq \tilde{h}_2(\tau,t) \otimes \tilde{h}_1(\tau,t) \tag{13.29}$$

Tuy nhiên, tính chất giao hoán áp dụng được cho các khối LTIV, nghĩa là:

$$\tilde{h}_1(\tau) \otimes \tilde{h}_2(\tau) = \tilde{h}_2(\tau) \otimes \tilde{h}_1(\tau) \tag{13.30}$$



Hình 13.6: Ví dụ đổi thứ tự các khối LTV
(tính giao hoán không áp dụng được đối với hệ thống LTV)

Tính phân phối

Ta có thể kiểm lại tính chất phân phối:

$$\tilde{x}(\tau) \otimes \left[\tilde{h}_1(\tau,t) + \tilde{h}_2(\tau,t)\right] = \tilde{x}(\tau) \otimes \tilde{h}_1(\tau,t) + \tilde{x}(\tau) \otimes \tilde{h}_2(\tau,t) \tag{13.31}$$

Tương đương với:

$$\tilde{h}_1(\tau,t) \otimes \left[\tilde{h}_2(\tau,t) + \tilde{h}_3(\tau,t)\right] = \tilde{h}_1(\tau,t) \otimes \tilde{h}_2(\tau,t) + \tilde{h}_1(\tau,t) \otimes \tilde{h}_3(\tau,t) \tag{13.32}$$

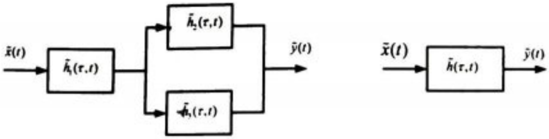
Đối với các hệ thống LTV.

Một ví dụ về việc đơn giản hoá các sơ đồ khối mô phỏng của các hệ thống LTV từ tính chất kết hợp và tính chất phân phối được cho ở hình 13.7, trong đó:

$$\tilde{h}(\tau,t) = \tilde{h}_1(\tau,t) \otimes \tilde{h}_2(\tau,t) + \tilde{h}_1(\tau,t) \otimes \tilde{h}_3(\tau,t) \tag{13.33}$$

Trong hình 13.7(b).

Đáng tiếc, việc đơn giản hoá dẫn đến hiệu quả tính toán trong các khối LTIV lại không áp dụng trực tiếp được cho các khối LTV. Sơ đồ khối có chứa hồi tiếp không thể đơn giản hoá được.



a) Các kết nối nối tiếp và song song b) Hệ thống tương đương

Hình 13.7: Các kết nối song song /nối tiếp và hệ thống tương đương

Ví dụ 13.1: Giả sử đầu vào của hệ thống LTV là một hàm mũ phức biên độ đơn vị có tần số là 1kHz. Nói cách khác:

$$\tilde{x}(t) = e^{j2000\pi t} \tag{13.34}$$

Tương ứng trong miền tần số là:

$$\tilde{X}(f) = \delta(f - 1000) \tag{13.35}$$

Hệ thống được định nghĩa bởi $\tilde{h}(\tau,t)$, được giả thiết là có một đáp ứng xung kim thay đổi theo thời gian.

$$\tilde{h}(\tau,t) = a_1(t).\delta(\tau) + a_2(t).\delta(\tau - T) + a_3(t).\delta(\tau - 2T) \tag{13.36}$$

Trong đó T là *1ms* và các suy hao thay đổi theo thời gian $a_i(t)$ được cho bởi:

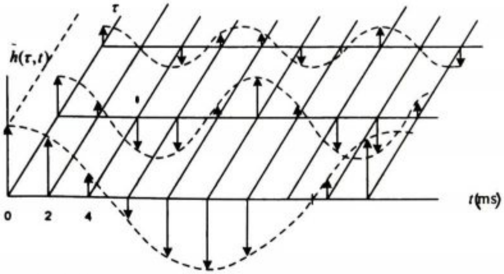
$$a_1(t) = 1 \times \cos(100\pi t) \tag{13.37}$$

$$a_2(t) = 0{,}7 \times \cos(200\pi t) \tag{13.38}$$

$$a_3(t) = 0{,}3 \times \cos(300\pi t) \tag{13.39}$$

Hình 13.8 minh họa 10 "***chớp ảnh***" của đáp ứng xung kim $\tilde{h}(\tau,t)$ cách nhau 2 *ms* bắt đầu tại $t = 0$ s. Ta có thể thấy rằng, hệ thống này là một hàm của cả thời gian trôi τ và thời điểm quan sát t. Biểu đồ loại này thường được sử dụng để mô tả loại kênh thay đổi theo thời gian của hệ thống truyền thông vô tuyến.

Đáp ứng tần số thay đổi theo thời gian của kênh này có được bằng cách thực hiện biến đổi Fourier 2 chiều được định nghĩa bởi (13.8). Thế đáp ứng xung kim được định nghĩa bởi (13.36) vào (13.8) ta được:



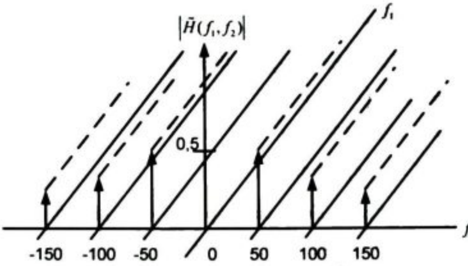
Hình 13.8: Minh họa $\tilde{h}(\tau,t)$ là một hàm của cả τ và t

$$\tilde{H}(f_1,f_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\cos(100\pi t)\times\delta(\tau)\times e^{-j2\pi f_1\tau}\times e^{-j2\pi f_2 t}\,d\tau dt$$
$$+\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}0{,}7\times\cos(200\pi t)\times\delta(\tau-T)\times e^{-j2\pi f_1\tau}\times e^{-j2\pi f_2 t}\,d\tau dt \qquad (13.40)$$
$$+\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}0{,}3\times\cos(300\pi t)\times\delta(\tau-2T)\times e^{-j2\pi f_1\tau}\times e^{-j2\pi f_2 t}\,d\tau dt$$

Sau đó áp dụng tính chất chọn lọc và điều chế của biến đổi Fourier, tìm được hàm truyền đạt:

$$\tilde{H}(f_1,f_2)=\frac{1}{2}\delta(f_2\pm 50)+\frac{0{,}7}{2}\delta(f_2\pm 100)e^{-j2\pi f_1 T}$$
$$+\frac{0{,}3}{2}\delta(f_2\pm 150)e^{-j4\pi f_1 T} \qquad (13.41)$$

Đáp ứng độ lớn của nó được cho ở hình 13.9. Lưu ý rằng, khác với hệ thống LTIV, hàm truyền đạt của hệ thống LTV phụ thuộc vào cả hai tần số f_1 và f_2.



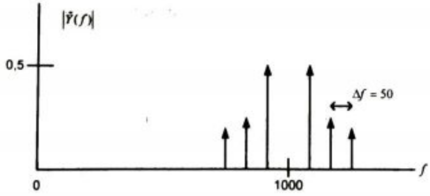
Hình 13.9: Biến đổi Fourier của đáp ứng xung kim thay đổi theo thời gian

Đầu ra của hệ thống được định nghĩa bởi (13.10). Thay (13.35) và (13.41) vào (13.10) ta được:

$$\tilde{Y}(f)=\frac{1}{2}\int_{-\infty}^{\infty}\delta[(f-f_1)\pm 50].\delta(f_1-1000)df_1$$
$$+\frac{0{,}7}{2}\int_{-\infty}^{\infty}\delta[(f-f_1)\pm 100].\delta(f_1-1000)e^{-j2\pi f_1 T}df_1 \qquad (13.42)$$
$$+\frac{0{,}3}{2}\int_{-\infty}^{\infty}\delta[(f-f_1)\pm 150]\delta(f_1-1000)e^{-j4\pi f_1 T}df_1$$

Đã sử dụng tính chất dịch của hàm delta, lấy tích phân ta được:

$$\tilde{Y}(f)=\frac{1}{2}\delta(f-1000\pm 50)+\frac{0{,}7}{2}\delta(f-1000\pm 100)e^{-j2000\pi T}$$
$$+\frac{0{,}3}{2}\delta(f-1000\pm 150)e^{-j4000\pi T} \qquad (13.43)$$



Hình 13.10: Đầu ra hệ thống

Đáp ứng của hệ thống thay đổi theo thời gian đối với đầu vào đã được giả định được minh họa ở hình 13.10. Ta thấy rằng, đầu vào là một đơn mang tại tần số 1000 Hz bị dịch tần và bị trải rộng ra trong miền tần số do đáp ứng xung kim kênh biến đổi theo thời gian gây ra. Ảnh hưởng của việc mở rộng phổ (trải phổ) sẽ được sáng tỏ ở phần sau.

13.3. Mô hình quá trình ngẫu nhiên

Trong nhiều hệ thống thay đổi theo thời gian, các đặc tính của hệ thống thay đổi một cách ngẫu nhiên là một hàm của thời gian. Ví dụ thay đổi trong các đặc tính phần cứng (do linh kiện bị lão hóa), và các thay đổi trong các đặc tính của kênh vô tuyến (chẳng hạn do các thay đổi trong các điều kiện mô trường khí quyển). Những thay đổi theo thời gian này thường được mô hình hóa bởi các quá trình ngẫu nhiên. Nếu mô hình cơ bản là tuyến tính, thì thay đổi ngẫu nhiên của đặc tính hệ thống là một hàm của thời gian có thể được xử lý bằng cách coi đáp ứng xung kim $\tilde{h}(\tau,t)$ là một quá trình ngẫu nhiên theo thời gian t. Giải pháp như vậy được sử dụng mở rộng để mô hình hóa cho các kênh vô tuyến di động. Chi tiết hóa cho các loại mô hình kênh khác nhau sẽ được đề cập ở các chương sau, tại đây giới thiệu vắn tắt nhằm triển khai mô hình mô phỏng chung cho các hệ thống thay đổi ngẫu nhiên theo thời gian. (trong phần này, ta sử dụng các hàm chữ nhỏ của thời gian để biểu thị cho các quá trình ngẫu nhiên mặc dù theo lệ thường các chữ hoa được dùng để biểu thị các biến ngẫu nhiên và các quá trình ngẫu nhiên. Các chữ cái hoa được sử dụng để biểu thị các biến đổi Fourier).

Ví dụ đơn giản: Xét một hệ thống có đầu ra là một phiên bản của đầu vào bị suy hao và bị trễ, trong đó suy hao thay đổi ngẫu nhiên là một hàm của thời gian. Hệ thống được đặc trưng hóa bởi một đáp ứng xung kim:

$$\tilde{h}(\tau,t) = a(t)\delta(\tau - t_0) \tag{13.44}$$

Trong đó $a(t)$ là suy hao thay đổi theo thời gian và t_0 là trễ. Quan hệ vào/ra được cho bởi:

$$\tilde{y}(\tau,t) = \int_{-\infty}^{\infty} \tilde{h}(\tau,t)\tilde{x}(t-\tau)d\tau = a(t)\tilde{x}(t-t_0) \tag{13.45}$$

Suy hao *a(t)* được mô hình hóa là một quá trình ngẫu nhiên *dừng* hoặc *không dừng*. Mô hình thích hợp hơn cả là một quá trình ngẫu nhiên dừng, vì mô hình này dẫn đến hàm tự tương quan và mật độ phổ công suất. Việc mô hình hóa *a(t)* là một quá trình dừng vẫn cho phép thể hiện được mô hình đáp ứng xung kim thay đổi theo thời gian.

Nếu đưa $\tilde{x}(t)$ vào hệ thống là một quá trình ngẫu nhiên dừng, thì hàm tự tương quan và mật độ phổ công suất của đầu ra là:

$$R_{\tilde{y}\tilde{y}}(\alpha) = E\left\{\tilde{y}^*(t)\tilde{y}(t+\alpha)\right\} \tag{13.46}$$

$$R_{\tilde{y}\tilde{y}}(\alpha) = E\left\{\tilde{a}^*(t).\tilde{x}^*(t-t_0).\ \tilde{a}(t+\alpha).\tilde{x}(t-t_0+\alpha)\right\} \tag{13.47}$$

Giả sử quá trình đầu vào $\tilde{x}(t)$ và suy hao $\tilde{a}(t)$ là *không tương quan nhau*, thì hàm tự tương quan đầu ra rút gọn về dạng đơn giản:

$$R_{\tilde{y}\tilde{y}}(\alpha) = R_{\tilde{a}\tilde{a}}(\alpha).R_{\tilde{x}\tilde{x}}(\alpha) \tag{13.48}$$

Bằng cách biến đổi Fourier của hàm tự tương quan trên, ta được mật độ phổ công suất PSD của đầu ra là:

$$S_{\tilde{y}\tilde{y}}(f) = S_{\tilde{a}\tilde{a}}(f) \otimes S_{\tilde{x}\tilde{x}}(f) \tag{13.49}$$

Phép tích chập trong phương trình trên dẫn đến dịch phổ và trải phổ.

Nếu đầu vào là hàm số mũ phức pha ngẫu nhiên:

$$\tilde{x}(t) = A e^{j(2\pi f_0 t+\theta)},\ \theta \text{ được phân bố đều trên khoảng } [-\pi, \pi]$$

thì tự tương quan đầu vào là:

$$R_{\tilde{x}\tilde{x}}(\alpha) = A^2 e^{j2\pi f_0 \alpha} \tag{13.50}$$

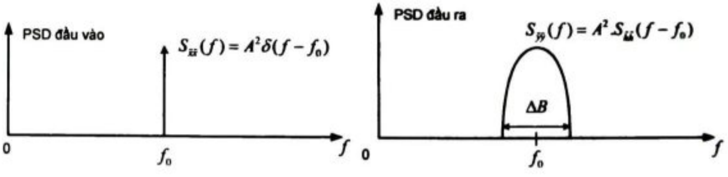
Từ đó mật độ phổ công suất PSD đầu vào là:

$$S_{\tilde{x}\tilde{x}}(f) = A^2\delta(f-f_0) \tag{13.51}$$

Mật độ phổ công suất PSD đầu ra là:

$$S_{\tilde{y}\tilde{y}}(f) = A^2 S_{\tilde{h}\tilde{h}}(f-f_0) \tag{13.52}$$

Như được minh họa trong hình 13.11.



Hình 13.11: Trải phổ trong hệ thống thay đổi theo thời gian

Hệ thống LTIV sẽ tạo ra một mật độ phổ công suất đầu ra dạng:

$$S_{\tilde{y}\tilde{y}}(f) = A^2.\delta(f-f_0)\left|.H(f_0)\right|^2 \tag{13.53}$$

Sự khác nhau căn bản giữa hệ thống LTV và hệ thống LTIV là trải phổ ở hệ thống LTV.

Đối với hệ thống LTV, trong mô phỏng phải đặc biệt lưu ý tốc độ lấy mẫu, phải tăng ít nhất là bằng hai lần độ rộng băng tần giãn độ rộng băng do trải phổ gây ra.

Phiên bản tổng quát hơn của mô hình thay đổi theo thời gian coi $\tilde{h}(\tau,t)$ là một quá trình ngẫu nhiên dừng theo *t* có hàm tự tương quan:

$$R_{\tilde{h}\tilde{h}}(\tau_1,\tau_2,\alpha)=E\left\{\tilde{h}^*(\tau_1,t)\tilde{h}(\tau_2,t+\alpha)\right\} \tag{13.54}$$

Mô hình được sử dụng phổ biến nhất cho $\tilde{h}(\tau,t)$ là quá trình Gausơ dừng trung bình không dẫn đến hàm mật độ xác xuất Rayleigh cho $\left|\tilde{h}(\tau,t)\right|$. Trong mô hình này thường giả sử rằng $\tilde{h}(\tau_1,t)$ và $\tilde{h}(\tau_2,t)$ không tương quan nhau khi $\tau_1 \neq \tau_2$, nói cách khác:

$$R_{\tilde{h}\tilde{h}}(\tau_1,\tau_2,\alpha)=E\left\{\tilde{h}^*(\tau_1,t).\tilde{h}(\tau_2,t+\alpha)\right\}=R_{\tilde{h}\tilde{h}}(\tau_1,\alpha).\delta(\tau_1-\tau_2) \tag{13.55}$$

Khi này, hàm tự tương quan đầu ra hệ thống là:

$$R_{\tilde{y}\tilde{y}}(\alpha)=E\left\{\tilde{y}^*(t)\tilde{y}(t+\alpha)\right\} \tag{13.56}$$

Thay $\tilde{y}(t)$ vào ta được:

$$R_{\tilde{y}\tilde{y}}(\alpha)=E\left\{\int_{-\infty}^{\infty}\tilde{h}^*(\tau_1,t)\tilde{x}^*(t-\tau_1)d\tau_1\times\int_{-\infty}^{\infty}\tilde{h}(\tau_2,t+\alpha)\tilde{x}(t+\alpha-\tau_2)d\tau_2\right\} \tag{13.57}$$

Đổi thứ tự phép lấy kỳ vọng và tích phân dẫn đến:

$$R_{\tilde{y}\tilde{y}}(\alpha)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}E\left\{\tilde{h}^*(\tau_1,t)\tilde{h}(\tau_2,t+\alpha)\times\tilde{x}^*(t-\tau_1)\tilde{x}(t+\alpha-\tau_2)\right\}d\tau_1d\tau_2 \tag{13.58}$$

Giả sử $\tilde{x}(t)$ và $\tilde{h}(t)$ độc lập, đạt được:

$$R_{\tilde{y}\tilde{y}}(\alpha)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left\{E\left[\tilde{h}^*(\tau_1,t)\tilde{h}(\tau_2,t+\alpha)\right]E\left[\tilde{x}^*(t-\tau_1)\tilde{x}(t+\alpha-\tau_2)\right]\right\}d\tau_1d\tau_2 \tag{13.59}$$

Thừa nhận rằng 2 phép kỳ vọng là các hàm tự tương quan và dựa vào (13.55) cho ta dạng đơn giản:

$$R_{\tilde{y}\tilde{y}}(\alpha)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}R_{\tilde{h}\tilde{h}}(\tau_1,\alpha)\delta(\tau_1-\tau_2)R_{\tilde{x}\tilde{x}}(\alpha+\tau_1-\tau_2)d\tau_1d\tau_2 \tag{13.60}$$

Thực hiện lấy tích phân theo τ_2 và dùng tính chất chọn lọc ta được:

$$R_{\tilde{y}\tilde{y}}(\alpha)=R_{\tilde{x}\tilde{x}}(\alpha)\int_{-\infty}^{\infty}R_{\tilde{h}\tilde{h}}(\tau_1,\alpha)d\tau_1 \tag{13.61}$$

Có thể được biểu diễn bởi:

$$R_{\tilde{y}\tilde{y}}(\alpha)=R_{\tilde{x}\tilde{x}}(\alpha)\overline{R}_{\tilde{h}\tilde{h}}(\alpha) \tag{13.62}$$

Trong đó:

$$\overline{R}_{\tilde{h}\tilde{h}}(\alpha)=\int_{-\infty}^{\infty}R_{\tilde{h}\tilde{h}}(\tau_1,\alpha)d\tau_1 \tag{13.63}$$

Mật độ phổ công suất đầu ra có thể tìm được bằng cách biến đổi Fourier của (13.62) dẫn đến tích chập:

$$S_{\tilde{y}\tilde{y}}(f) = S_{\tilde{x}\tilde{x}}(f) \otimes \overline{S}_{\tilde{h}\tilde{h}}(f) \tag{13.64}$$

Trong đó ⊗ là tích chập. Lưu ý rằng $\overline{S}_{\tilde{h}\tilde{h}}(f)$ là mật độ phổ công suất "*được trung bình hóa*" và là biến đổi Fourier của hàm tự tương quan "*được trung bình hóa*" được định nghĩa trong (13.63).

Lưu ý rằng, mật độ phổ công suất đầu ra là tích chập của mật độ phổ công suất đầu vào và mật độ phổ công suất "*được trung bình hóa*" của quá trình ngẫu nhiên mà nó mô hình hóa cho các thay đổi theo thời gian. Ngược lại, quan hệ về mật độ phổ công suất của một hệ thống LTIV được cho bởi:

$$S_{\tilde{y}\tilde{y}}(f) = S_{\tilde{x}\tilde{x}}(f)\left|H(f)\right|^2 \tag{13.65}$$

Một lần nữa khi đầu vào là một đơn mang, thì hệ thống LTIV tạo ra đầu ra đơn mang tại cùng tần số đầu vào, trong khi đó đầu ra của hệ thống LTV có thể bị dịch phổ tần và trải rộng phổ tần.

13.4. Mô hình mô phỏng cho hệ thống LTV

Cho trước một mô tả một hệ thống LTV ở dạng của đáp ứng xung kim $\tilde{h}(\tau,t)$, thì ta có thể rút ra mô hình mô phỏng bằng cách sử dụng định lý lấy mẫu với giả thiết tín hiệu đưa vào kênh có băng tần giới hạn. Ta bắt đầu từ quan hệ vào/ra được cho bởi tích chập:

$$\tilde{y}(t) = \int_{-\infty}^{\infty} \tilde{h}(\tau,t)\tilde{x}(t-\tau)d\tau$$

Và sử dụng định lý lấy mẫu để biểu diễn đầu vào theo các giá trị mẫu của nó.

Từ định lý lấy mẫu cho thấy, một tín hiệu thông thấp $\tilde{w}(\tau)$ có băng tần được giới hạn B Hz được biểu diễn theo các giá trị mẫu của nó là:

$$\tilde{w}(\tau) = \sum_{n=-\infty}^{\infty} \tilde{w}(nT)\frac{\sin\left(2\pi B(\tau - nT)\right)}{2\pi B(\tau - nT)} \tag{13.66}$$

Trong đó *1/T* là tốc độ lấy mẫu bằng với tốc độ Nyquist *2B*. Chọn tốc độ lấy mẫu nhỏ nhất là 2B để giảm thiểu tải tính toán cho mô hình mô phỏng. Sử dụng cách biểu diễn trên, với $\tilde{x}(t-\tau) = \tilde{w}(\tau)$, ta có thể thay $\tilde{x}(t-\tau)$ trong tích chập bằng

$$\tilde{x}(t-\tau) = \sum_{n=-\infty}^{\infty} \tilde{x}(t-nT)\frac{\sin\left(2\pi B(\tau - nT)\right)}{2\pi B(\tau - nT)} \tag{13.67}$$

Dẫn đến:

$$\begin{aligned}\tilde{y}(\tau) &= \int_{-\infty}^{\infty} \tilde{h}(\tau,t)\left\{\sum_{n=-\infty}^{\infty} \tilde{x}(t-nT)\frac{\sin\left(2\pi B(\tau - nT)\right)}{2\pi B(\tau - nT)}\right\}d\tau \\ &= \sum_{n=-\infty}^{\infty} \tilde{x}(t-nT)\int_{-\infty}^{\infty} \tilde{h}(\tau,t)\left\{\frac{\sin\left(2\pi B(\tau - nT)\right)}{2\pi B(\tau - nT)}\right\}d\tau\end{aligned} \tag{13.68}$$

Vì vậy, ta có thể viết:

$$\tilde{y}(t) = \sum_{n=-\infty}^{\infty} \tilde{x}(t-nT)\tilde{g}_n(t) \tag{13.69}$$

Trong đó:

$$\tilde{g}_n(t) = \int_{-\infty}^{\infty} \tilde{h}(\tau,t)\left\{\frac{\sin\left(2\pi B(\tau-nT)\right)}{2\pi B(\tau-nT)}\right\}d\tau \tag{13.70}$$

Các phương trình (13.69) và (13.70) định nghĩa mô hình mô phỏng cho các hệ thống thay đổi theo thời gian.

Mô hình đường trễ nhánh TDL

Mô hình được cho trong (13.69) có thể được thực hiện ở dạng đường trễ nhánh TDL như được cho ở hình 13.12, với các hàm độ lợi nhánh được định rõ bởi (13.70). ở dạng tổng quát, các hàm độ lợi nhánh là các quá trình ngẫu nhiên và chúng sẽ bị tương quan nhau, nghĩa là $\tilde{g}_n(t)$ và $\tilde{g}_m(t)$ sẽ bị tương quan nhau.

Đơn giản hóa mô hình đường trễ rẽ nhánh TDL

Mô hình được cho ở trên có thể được đơn giản hóa theo nhiều cách sử dụng các giả định và các phép tính xấp xỉ.

Thứ nhất, thường giả định quá trình độ lợi nhánh là *không tương quan*. Chúng cũng được xấp xỉ bởi:

$$\tilde{g}_n(t) \approx T\tilde{h}\left(nT,t\right) \tag{13.71}$$

Trong phép tính gần đúng này, các hàm độ lợi nhánh biểu diễn các giá trị mẫu của đáp ứng xung kim thay đổi theo thời gian $\tilde{h}(\tau,t)$, trong đó việc lấy mẫu được thực hiện trong biến đáp ứng xung kim.

Thứ hai, phép tính gần đúng thứ 2 bao gồm việc *cắt đáp ứng xung kim*. Nếu:

$$E\left\{\left|\tilde{h}(\tau,t)\right|^2\right\} \to 0, \quad |\tau| \geq mT$$

Thì phép lấy tổng trong (13.69) được cắt giảm còn *2m+1* số hạng như sau:

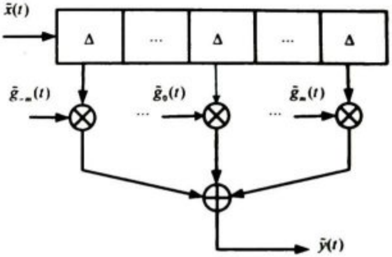
$$\sum_{n=-\infty}^{\infty} \tilde{x}(t-nT)\tilde{g}_n(t) \approx \sum_{n=-m}^{m} \tilde{x}(t-nT).\tilde{g}_n(t) \tag{13.72}$$

Và mô hình đường trễ rẽ nhánh chỉ có hữu hạn các nhánh như được thấy trong hình 13.12. Tổng số nhánh nên được giữ ở mức nhỏ nhất để tối đa hóa hiệu quả tính toán của mô hình.

Cuối cùng, nếu hệ thống là bất biến, thì $\tilde{h}(\tau,t) = \tilde{h}(\tau)$, và các *độ lợi nhánh trở thành hằng số:*

$$\tilde{g}_n(t) = \tilde{g}_n \approx T\tilde{h}(nT)$$

Nói cách khác, các độ lợi nhánh chỉ là các giá trị mẫu của đáp ứng xung kim của hệ thống LTIV, và mô hình đường trễ rẽ nhánh rút gọn thành một mô hình bất biến xung kim, hoặc bộ lọc FIR thực hiện tích chập trong miền thời gian (xem hình 5.4 chương 5).

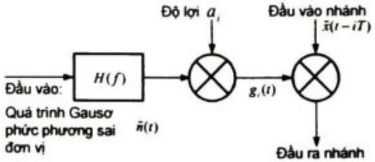


Hình 13.12: Mô hình đường trễ nhánh TDL

Theo một khía cạnh khác của TDL cũng đáng được quan tâm. Mô hình TDL được cho ở hình 13.12 có tín hiệu vào/ra $\tilde{x}(t)/\tilde{y}(t)$ liên tục theo thời gian. Tuy nhiên, trong mô phỏng ta phải dùng các giá trị mẫu của $\tilde{x}(t)$ và $\tilde{y}(t)$. Việc lấy mẫu thường được thực hiện với tốc độ lấy mẫu từ 8 đên 16 lần độ rộng băng tần, trong đó độ rộng băng tần phải tính đến ảnh hưởng của trải phổ do bản chất thay đổi theo thời gian của hệ thống gây ra. Các ảnh hưởng này được thấy trong biểu thức (13.52) và (13.64). Lưu ý rằng, tốc độ Nyquist là 2B đã được sử dụng để rút ra mô hình đường trễ rẽ nhánh TDL và khoảng cách nhánh là $T = 1/2B$, nó sẽ $>>T_s$, trong đó T_s là chu kỳ lấy mẫu cho các tín hiệu vào/ra. Tất nhiên, có thể rút ra mô hình đường trễ rẽ nhánh TDL có khoảng cách nhánh nhỏ hơn T_s, nhưng một mô hình như vậy không mang lại hiệu quả tính toán và không cần thiết để cải thiện độ chính xác.

Tạo các quá trình độ lợi nhánh

Các quá trình độ lợi nhánh là các quá trình ngẫu nhiên dừng có một hàm mật độ xác suất và mật độ phổ công suất cho trước. Mô hình đơn giản nhất cho các quá trình độ lợi nhánh là coi chúng là các quá trình Gausơ phức trung bình không, không tương quan có các phương sai khác nhau nhưng các mật độ phổ công suất như nhau. Trong trường hợp này, các quá trình độ lợi nhánh có thể được tạo ra bằng cách lọc các quá trình Gausơ trắng như được cho ở hình 13.13.



Hình 13.13: Tạo các quá trình độ lợi nhánh thứ i, $t = kT_s$

Hàm truyền đạt bộ lọc được chọn sao cho tạo ra mật độ phổ công suất mong muốn nghĩa là *H(f)* được chọn sao cho:

$$S_{\tilde{g}\tilde{g}}(f) = S_{\tilde{n}\tilde{n}}(f).\left|\tilde{H}(f)\right|^2 = \left|\tilde{H}(f)\right|^2 \tag{13.73}$$

Trong đó $S_{\tilde{n}\tilde{n}}(f)$ là mật độ phổ công suất của quá trình tạp âm đầu vào, được đặt bằng 1, và $S_{\tilde{g}\tilde{g}}(f)$ là mật độ phổ công suất cụ thể của các quá trình độ lợi nhánh. Độ lợi không đổi (tĩnh) a_i trong hình 13.13 giải thích cho các mức công suất khác nhau hoặc các phương sai của các nhánh khác nhau (lưu ý đến quan hệ giữa phương sai và công suất của quá trình ngẫu nhiên). Nếu mật độ phổ công suất của các độ lợi nhánh là khác nhau, thì dùng các bộ lọc khác nhau cho các nhánh khác nhau.

Có một số cách thực hiện bộ lọc định dạng phổ (cũng được xem là bộ lọc doppler trong các mô hình kênh). Bộ lọc FIR trong miền thời gian là cách phổ biến nhất vì các mật độ phổ công suất (doppler) làm chính nó phân tán phổ và thực hiện ở dạng lặp.

Khi tạo các quá trình độ lợi nhánh, cần lưu ý độ rộng băng tần của các quá trình độ lợi nhánh của các hệ thống thay đổi chậm theo thời gian sẽ nhỏ so với độ rộng băng tần hệ thống và độ rộng băng tần tín hiệu truyền qua hệ thống đó. Trong trường hợp này, bộ lọc độ lợi nhánh nên được thiết kế và thực hiện ở một tốc độ lấy mẫu thấp hơn. Phép nội suy có thể được dùng ở đầu ra bộ lọc để tạo các mẫu dày đặc hơn tại tốc độ phù hợp với tốc độ lấy mẫu của tín hiệu đến nhánh đó. Việc thiết kế bộ lọc ở tốc độ lấy mẫu cao hơn sẽ dẫn đến không hiệu quả tính toán cũng như tính ổn định.

13.5. Thực hiện mô phỏng trên Matlab

Ta kết thúc chương bằng 2 chương trình Matlab điển hình minh họa khái niệm được trình bày trong chương này.

13.5.1. Mô phỏng hiện tượng trải rộng Doppler

Trường hợp này minh họa trải phổ (trải rộng Doppler) xảy ra trong hệ thống thay đổi theo thời gian như được cho trong (13.64). Hệ thống được mô phỏng là một hệ thống đơn giản, đầu vào là một "*đơn mang*" thông dải dạng $x(t) = \cos[2\pi(f_0 + f_1)t]$ tương ứng với tín hiệu tương đương thông thấp là:

$$\tilde{x}(t) = e^{j2\pi f_1 t} \tag{13.74}$$

Đáp ứng xung kim tương đương thông thấp của hệ thống được giả định có dạng:

$$\tilde{h}(\tau,t) = \tilde{a}(t).\delta(t-\tau_0) \tag{13.75}$$

Là một *kênh toàn thông* có trễ τ_0 và một suy hao phức thay đổi theo thời gian $\tilde{a}(t)$. Suy hao được mô hình hóa là một quá trình ngẫu nhiên Gausơ trung bình không có mật độ phổ công suất là:

$$S_{\tilde{a}\tilde{a}}(f) = \frac{1}{(j2\pi f)^2 + B^2} \tag{13.76}$$

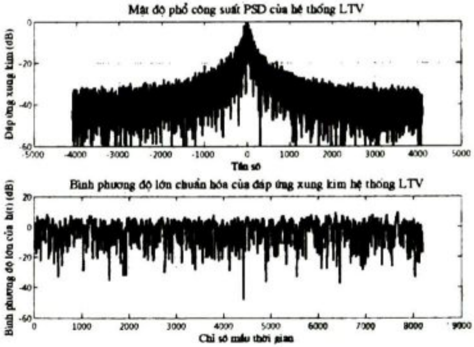
Trong đó B là độ rộng băng tần doppler. Đường bao phức đầu ra hệ thống thay đổi theo thời gian là:

$$\tilde{y}(t) = \tilde{a}(t).\tilde{x}(t-\tau_0) \tag{13.77}$$

Mô phỏng mô hình này bao gồm việc tạo ra các giá trị mẫu của đơn mang đầu vào và nhân với một quá trình Gausơ phức được lọc, trong đó bộ lọc được chọn để có được một hàm truyền đạt cho ta mật độ phổ công suất theo (13.76). Thấy rõ, phải có bộ lọc có hàm truyền đạt:

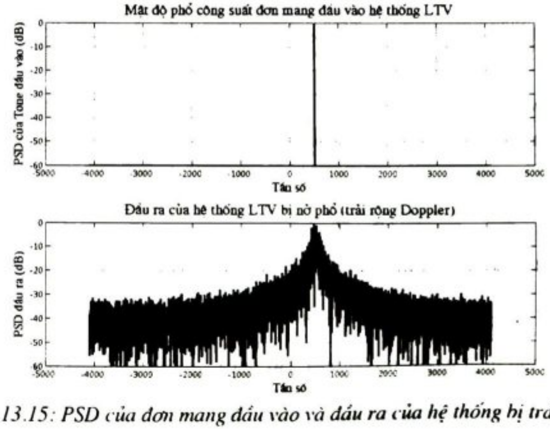
$$H(s) = \frac{1}{s+B} \tag{13.78}$$

Kết quả mô phỏng được minh họa với các giá trị tham số $f_l = 512$ Hz; $B = 64$ Hz; $\tau_0 = 0$; $f_s = 8.192$ mẫu /s và độ dài mô phỏng = 8.192 mẫu.



Hình 13.14 : Trình bày các quá trình Gausơ được lọc trong miền tần số (trên) và miền thời gian (dưới)

Các kết quả mô phỏng được cho ở hình 13.14 và 13.15. Mật độ phổ công suất của quá trình Gausơ được lọc và các giá trị của chuỗi Gausơ tương quan trong miền thời gian được tạo ra bởi bộ lọc hình 13.14. Đơn mang đầu vào được biểu diễn theo (13.74) và đầu ra của hệ thống thay đổi theo thời gian trong miền tần số được cho ở hình 13.15. Trải phổ do tính chất thay đổi theo thời gian của hệ thống được thấy ở khung dưới của hình 13.5. Mã chương trình Matlab thực hiện bài toán này được cho ở **NVD13_tiv1.m** trong phụ lục 13A.



Hình 13.15: PSD của đơn mang đầu vào và đầu ra của hệ thống bị trải rộng

13.5.2. Mô phỏng tính chọn lọc tần số và tính thay đổi theo thời gian

Trường hợp này ta minh họa 2 khía cạnh của kênh thay đổi theo thời gian: tính chọn lọc tần số và thay đổi theo thời gian. Trong trường hợp trên kênh là *toàn băng (allpass)* hay kênh không chọn lọc tần số (đôi khi được gọi là kênh phẳng). Tại đây, ta biến đổi mô hình hệ thống sao cho đáp ứng xung kim tương đương thông thấp phức có dạng:

$$\tilde{h}(\tau,t)=\tilde{a}_1(t).\delta(\tau-\tau_1)+\tilde{a}_2(t).\delta(\tau-\tau_2) \tag{13.79}$$

Thấy rõ, mô hình này là mô hình chọn lọc tần số bằng cách xét một phiên bản bất biến theo thời gian với:

$$\tilde{h}(\tau,t)\Big|_{\substack{\tilde{a}_1(t)=\tilde{a}_1\\ \tilde{a}_2(t)=\tilde{a}_2}}=\underbrace{\tilde{a}_1.\delta(\tau-\tau_1)+\tilde{a}_2.\delta(\tau-\tau_2)}_{\tilde{h}(\tau)} \tag{13.80}$$

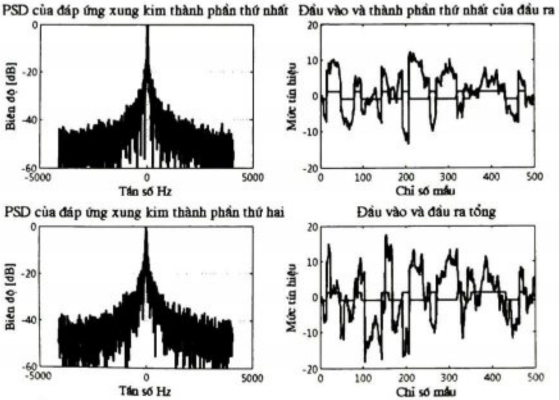
Dẫn đến hàm truyền đạt với giả thiết $\tau_1=0$, ta được

$$H(f)=\tilde{a}_1+\tilde{a}_2\,e^{-j2\pi f\tau_2} \tag{13.81}$$

Tính chọn lọc tần số được thể hiện ở chỗ hàm truyền đạt này có các giá trị khác nhau tại các tần số khác nhau.

Đầu ra tương đương thông thấp phức của hệ thống thay đổi theo thời gian được cho bởi (13.79) là:

$$\tilde{y}(t)=\tilde{a}_1(t).\tilde{x}(t-\tau_1)+\tilde{a}_2(t-\tau_2).\tilde{x}(\tau-\tau_2) \tag{13.82}$$



Hình 13.16: Ảnh hưởng của một thành phần và hai thành phần thay đổi theo thời gian

Nếu đưa vào hệ thống là tín hiệu BPSK, thì tính thay đổi theo thời gian của hệ thống làm suy hao và xoay pha tín hiệu BPSK là một hàm của thời gian. Bản chất chọn lọc tần số của kênh là do 2 thành phần bị trễ của đầu ra [phương trình (13.79) và (13.81)], biểu lộ ở dạng ISI. Cả

hai ảnh hưởng này đều được tường minh trong các kết quả mô phỏng được cho ở hình 13.16. Các kết quả mô phỏng có được bằng cách chạy chương trình Matlab **NVD13_tiv2.m** trong phụ lục 13A với các tham số: tốc độ ký hiệu BPSK = 512 bit/s, tốc độ lấy mẫu = 16 mẫu/bit, độ rộng băng tần Doppler cho $\tilde{a}_1(t), \tilde{a}_2(t) = 16Hz$, $\tau_1 = 0$ và $\tau_2 = 8$ mẫu (một nửa khoảng thời gian bit). Trong hình 13.16, nửa trên minh họa ảnh hưởng của một thành phần thay đổi theo thời gian và nửa dưới minh họa ảnh hưởng của 2 thành phần. Chỉ 500 mẫu đầu tiên được chỉ ra. (Lưu ý rằng: chương trình được cho trong phụ lục 13A cũng tạo ra và hiển thị PSD của đáp ứng xung kim thành phần đầu tiên. Như trong ví dụ trước, mã chương trình Matlab được cho trong phụ lục 13A tạo ra 4 đồ thị khác nhau. Để được rõ hơn, cần phải chạy chương trình nhiều lần với các giá trị khác nhau của tham số mô phỏng và quan sát, so sánh kết quả mô phỏng tương ứng với việc thay đổi giá trị tham số.

Chương 14

MÔ HÌNH HÓA VÀ MÔ PHỎNG KÊNH DẠNG SÓNG

14.1 Mở đầu

Từ quan điểm thiết kế và phân tích hiệu năng hệ thống, nếu dung hòa giữa tính phức tạp của hệ thống truyền thông với khả năng xử lý, giá thành của máy tính, thì bài toán thiết kế dựa vào mô phỏng là các công cụ khả thực xuyên suốt công nghiệp truyền thông và ngày càng phổ biến hơn. Thông qua việc dùng mô phỏng, cho phép nghiên cứu các đặc trưng hoạt động của hệ thống phức tạp hơn, thực tế hơn so với những hệ thống không dựa trên mô phỏng bởi lẽ ở các hệ thống không dựa vào mô phỏng tính phức tạp bị hạn chế để đảm bảo việc phân tích có thể kiểm soát được. Mô phỏng cho phép thay đổi một cách dễ dàng các thông số hệ thống và nhanh chóng đánh giá được các ảnh hưởng của việc thay đổi đó ở dạng hiển thị trực quan và tương tác của các kết quả mô phỏng. Ngoài ra, hiểu các kỹ thuật mô phỏng sẽ hỗ trợ cho chương trình nghiên cứu của bạn đọc đang làm việc trong lĩnh vực truyền thông.

Trong chương này, ta tập trung vào việc mô hình hóa và mô phỏng kênh dạng sóng nhằm lột tả hoạt động và ảnh hưởng của nó ở dạng các tham số đặc trưng. Các tham số này sẽ làm đầu vào cho các giải thuật thích ứng, làm cơ sở để thiết kế các hệ thống thông minh cho các nghiên cứu của ta sau này.

Điều gì gây khó khăn việc thiết kế các kênh truyền thông như kênh vô tuyến di động, gây ra nhiều mức độ nhiễu, méo và tạp âm. Kênh vô tuyến di động cũng là kênh thay đổi theo thời gian và chịu ảnh hưởng của pha đinh. Ngoài ra, một số kênh thay đổi đến mức khó đặc trưng hóa chúng được. Hơn nữa, các hệ thống thông tin vô tuyến thế hệ sau phải được thiết kế để hoạt động trên các các kênh vô tuyến toàn cầu với tính đa dạng của môi trường từ khu vực thành phố đến các địa hình đồi núi, trong các điều kiện thời tiết khác nhau. Giải pháp xây dựng các nguyên mẫu cho hệ thống đã được đề xuất và trường kiểm tra nguyên mẫu tại nhiều vùng trên toàn cầu, nhưng rất tốn kém và sẽ không khả thi ở các giai đoạn đầu của quá trình thiết kế hệ thống. Giải pháp có tính khả thi cao là tạo ra các mô hình thích hợp cho kênh và dựa vào *thiết kế khởi đầu* trên các mô hình này.

Nếu có thể, cho trước các mô hình kênh truyền thông là mô hình thống kê hoặc tất định, thì ít nhất ở các giai đoạn đầu của quá trình thiết kế hệ thống sử dụng các giải pháp phân tích để ước lượng hiệu năng. Ví dụ, nếu coi "*pha đinh*" trong kênh có hàm mật độ xác suất biên độ phân bố Rayleigh và tạp âm là Gausơ cộng, thì xác suất lỗi đối với hệ thống truyền thông nhị phân làm việc trên kênh này được biểu diễn là:

$$P_e = 1/2\overline{\gamma_b} \tag{14.1}$$

Trong đó $\overline{\gamma_b}$ là giá trị "*trung bình*" của SNR tại đầu vào máy thu. Biểu thức này được dùng để xác định nhiều tham số như công suất máy phát để đảm bảo xác suất lỗi cho trước. Tuy nhiên, khi xây dựng hệ thống thực tế, những hạn chế về tính khả thi phải được xét đến như

bộ lọc không lý tưởng, bộ khuếch đại phi tuyến. Các ảnh hưởng này rất khó để đặc trưng hóa chúng theo cách phân tích và hầu hết phải dùng đến mô phỏng hoặc kết hợp mô phỏng với phân tích giải tích. Vì vậy, lập mô hình và mô phỏng đóng vai trò trung tâm trong thiết kế các hệ thống truyền thông. Hai chủ đề bao trùm trong chương này là: phương pháp luận và giải pháp mô phỏng cho các kênh truyền thông vô tuyến.

Tài liệu về đo đạc di động và các kênh vô tuyến khác rất phong phú và thay đổi. Với mục đích mô phỏng, ta thường dựa vào các mô hình thống kê được rút ra từ đo đạc. Một số ví dụ về các mô hình được dùng để thiết kế và ước lượng hiệu năng các hệ thống truyền thông di động thế hệ thứ 2 và 3 được trình bày trong chương này.

14.1.1. Các mô hình kênh truyền thông

Trong khi "*kênh*" truyền thông biểu diễn môi trường vật lý giữa máy phát và mày thu, thì "*mô hình kênh*" trình bày quan hệ vào/ra của kênh ở dạng toán học hoặc thuật toán. Mô hình này có thể được rút ra từ đo đạc hoặc dựa trên lý thuyết truyền lan vật lý. Các mô hình dựa trên đo đạc dẫn đến việc đặc trưng hóa theo kinh nghiệm cho kênh trong miền tần số hoặc thời gian, và bao hàm các mô tả thống kê dưới dạng các biến ngẫu nhiên hoặc quá trình ngẫu nhiên. Các tham số của các phân bố cơ bản và các mật độ phổ công suất thường được ước tính từ số liệu đo. *Trong khi* các mô hình dựa trên đo đạc vẫn có tin tưởng cao về tính hợp lệ và thường là những mô hình hữu dụng nhất để thiết kế thành công, *thì* thường tỏ ra khó dùng và khó tổng quát hóa trừ phi phép đo mở rộng được tập hợp trên các môi trường phù hợp. Ví dụ, rất khó sử dụng các kết quả đo từ một vùng thành phố này để *đặc tính hóa mô hình* cho vùng thành phố khác *trừ khi* một lượng lớn dữ liệu được tập hợp rộng về không gian và đa dạng về địa hình, và cần có lý thuyết cơ bản đủ mạnh để giải thích ngoại suy mô hình đó cho vị trí mới.

Việc triển khai mô hình toán cho truyền lan tín hiệu cần có hiểu biết sâu về các hiện tượng vật lý cơ bản. Ví dụ: để mô hình hóa cho kênh vô tuyến tầng điện ly, phải hiểu rõ truyền lan sóng vô tuyến. Tương tự, phải có hiểu biết căn bản về kiến thức quang học mới triển khai được các mô hình cho các sợi quang đơn mode và đa mode.

Một trong những thách thức khi lập mô hình kênh là chuyển mô hình truyền lan vật lý chi tiết thành dạng phù hợp để mô phỏng. Từ viễn cảnh vật lý, các mô hình toán có thể quá chi tiết hoặc có thể không ở dạng phù hợp để mô phỏng. Ví dụ, mô hình toán cho kênh vô tuyến có thể có dạng phương trình Maxwell. Mô hình này phải được đơn giản và được chuyển sang dạng thuận tiện như: hàm truyền đạt hoặc đáp ứng xung kim trước khi mô phỏng. May thay, có phần dễ dàng hơn so với việc rút ra các mô hình vật lý cơ bản và định rõ các tham số của những mô hình này. Một khi rút ra được mô hình vật lý và định rõ các giá trị tham số, việc chuyển mô hình vật lý sang mô hình mô phỏng (thuật toán) thường dễ ràng.

14.1.2. Mô phỏng các kênh truyền thông

Các kênh truyền thông vật lý như: cáp đồng, ống dẫn sóng, không gian tự do và sợi quang thường thể hiện tính tuyến tính. Một số kênh như kênh vô tuyến di động, tỏ ra vừa có tính tuyến tính vừa có tính thay đổi ngẫu nhiên theo thời gian LTV. Mô hình mô phỏng cho những kênh này thuộc một trong hai loại sau:

1. Mô hình hàm truyền đạt cho kênh bất biến theo thời gian. Ví dụ: cáp đồng, cáp quang và không gian tự do. Khi này, ta coi kênh có bản chất tĩnh (nghĩa là, kênh có đáp ứng xung kim bất biến theo thời gian) tạo ra một đáp ứng tần số cụ thể bởi lẽ trễ không đổi trong kênh. Hàm truyền đạt của kênh bất biết theo thời gian là "*phẳng*" nếu đặt vào kênh một tín hiệu có độ rộng băng thông, thì đáp ứng của kênh lên tín hiệu đó là không đổi. Kênh được gọi là "*chọn lọc tần số*" nếu tín hiệu vào có một độ rộng băng thông mà trên đó tác động của kênh với các mức độ khác nhau tại các tần số khác nhau trong độ rộng băng tín hiệu hay nói cách khác đáp ứng tần số của kênh trong vùng tần số tín hiệu là thay đổi.

2. Mô hình đường trễ rẽ nhánh TDL cho kênh thay đổi theo thời gian. Một ví dụ quan trọng là kênh vô tuyến di động. Với mô hình kênh này, kênh thường được coi là thay đổi theo thời gian. Nếu như kênh thay đổi trong khoảng thời gian nhỏ nhất của tín hiệu đặt vào, thì kênh được gọi là "*pha đinh nhanh*". Nếu như kênh vẫn không đổi trong nhiều ký hiệu liên tiếp của nguồn tín hiệu đặt vào, thì kênh được gọi là kênh "*pha đinh chậm*" và khi này được coi như loại (1).

Các mô hình hàm truyền đạt có thể được mô phỏng trong miền thời gian hoặc miền tần số sử dụng bộ lọc đáp ứng xung kim hữu hạn (FIR) hoặc đáp ứng xung kim vô hạn (IIR). Các mô hình kinh nghiệm ở dạng: đáp ứng được đo hoặc đáp xung kim được tổng hợp hoặc đáp ứng tần số thường được mô phỏng bằng cách dùng kỹ thuật FIR. Biểu thức giải tích cho hàm truyền đạt là dễ dàng hơn để mô phỏng sử dụng kỹ thuật IIR. Các bô lọc IIR và FIR đã được thảo luận chi tiết trong chương 5.

Các mô hình mô phỏng cho các kênh (pha đinh) thay đổi ngẫu nhiên theo thời gian dạng mô hình đường trễ rẽ nhánh TDL có các độ lợi nhánh và các trễ là các quá trình ngẫu nhiên. Nếu biết trước mô hình quá trình ngẫu nhiên cho các thay đổi theo thời gian cơ bản (pha đinh), thì có thể rút ra được các thuộc tính của quá trình độ lợi nhánh và được mô phỏng bằng các kỹ thuật được đề cập ở chương 13. Nếu kênh được giả định là kênh thay đổi *chậm* theo thời gian sao cho các điều kiện kênh không thay đổi trên nhiều ký hiệu phát, thì ta có thể dùng chớp ảnh (nghĩa là, đáp ứng xung kim tĩnh) của kênh để mô phỏng. Điều này có thể được lặp khi các điều kiện kênh thay đổi. Bằng cách lặp các mô phỏng cho nhiều điều kiện kênh, ta có thể suy ra hiệu năng hệ thống trong các chu kỳ thời gian dài dùng các phép đo hiệu năng như xác suất ngừng hoạt động như đã đươc đề cập ở chương 11.

14.1.3 Mô hình kênh rời rạc

Trọng tâm của chương này là các mô hình kênh dạng sóng để trình bày tương tác vật lý giữa dạng sóng tín hiệu phát và kênh. Các mô hình kênh dạng sóng được lấy mẫu tại tần số lấy mẫu phù hợp. Các mẫu được xử lý thông qua mô hình mô phỏng. Kỹ thuật khác thường hiệu quả hơn cho vài ứng dụng là biểu diễn kênh bằng một số hữu hạn các trạng thái. Tiến triển theo thời gian, trạng thái kênh thay đổi theo tập xác suất chuyển dịch, thì có thể định nghĩa kênh bởi một chuỗi Markov. Mô hình kênh kết quả hầu như có dạng mô hình Markov ẩn (HMM). Giả sử HMM được xây dựng chính xác, thì việc mô phỏng dựa trên HMM cho phép đạc tính hóa chính xác hiệu năng hệ thống truyền thông với hiệu quả tính toán cao. Các mô hình kênh rời rạc và các HMM là chủ đề của chương 15.

14.1.4 Phương pháp luận mô phỏng hiệu năng hệ thống truyền thông

Mô phỏng hiệu năng hệ thống truyền thông hoạt động trên kênh bất biến theo thời gian là khá đơn giản. Khi này, kênh được coi là một khối tuyến tính bất biến LTIV. Mặt khác, kênh thay đổi theo thời gian yêu cầu phải xét cụ thể. Phương pháp luận được dùng phụ thuộc vào mục đích mô phỏng và phụ thuộc vào kênh thuộc loại thay đổi *nhanh* hay *chậm* theo tín hiệu và các phân hệ đang được mô phỏng. Nhân tố quan trọng khác là quan hệ giữa độ rộng băng thông của tín hiệu và độ rộng băng thông của kênh. Mức độ phức tạp của mô hình kênh là hàm của các đặc tính thời gian và các đặc tính tần số của cả nguồn tín hiệu và kênh.

Nét đặc thù của chương

Phần đầu được dành để triển khai mô hình cho kênh truyền thông, bắt đầu bằng các mô hình hàm truyền đạt đơn giản cho kênh "*ống dẫn*" hoặc "*kim loại*". Kênh này bao gồm cáp xoắn, cáp, ống dẫn sóng và cáp quang, là các kênh tuyến tính bất biến theo thời gian LTIV vì vậy mô hình đáp ứng xung kim tĩnh và hàm truyền đạt là đủ. Sau đó, ta xét các mô hình kênh cho kênh vô tuyến không gian tự do, là tuyến tính nhưng có thể thay đổi theo thời gian.

Phần sau đề cập việc mô phỏng các kênh truyền thông, nhấn mạnh việc thực thi các mô hình đường trễ rẽ nhánh TDL cho các kênh thay đổi theo thời gian một cách ngẫu nhiên. Triển khai ba mô hình đường trễ rẽ nhánh TDL khác nhau theo mức độ tăng dần về mức độ phức tạp và dung lượng.

Kết thúc chương là mô tả phương pháp luận để mô phỏng hiệu năng các hệ thống truyền thông hoạt động trên các kênh pha đinh. Qua chương này, các kênh truyền thông mặt đất và di động sẽ được nhấn mạnh, bởi lẽ những kênh này thể hiện phần lớn các thách thức trong quá trình mô hình hóa và mô phỏng kênh, và cũng là nguyên nhân của sự quan tâm cao trong truyền thông vô tuyến.

14.2. Kênh hữu tuyến và ống dẫn sóng

Các hệ thống truyền thông điện dùng nhiều loại dây dẫn khác nhau như cáp xoắn đôi, cáp đồng trục. Các kênh này được đặc trưng bởi mô hình mạch điện RLC và đặc tính truyền đạt vào/ra được mô hình hoá bằng hàm truyền đạt. Dùng hàm truyền đạt khi lập mô hình mô phỏng. Dễ dàng đo đáp ứng tần số với sự thay đổi độ dài cáp và rút ra mô hình hàm truyền đạt dựa trên kết quả đo. Có thể cần có mạng cáp lớn để định nghĩa kênh bằng cách dùng một số biến ngẫu nhiên để đặc trưng hóa các tham số hàm truyền đạt hay đáp ứng xung *tĩnh*. Khi này, kênh được xem là bất biến theo thời gian, vì vậy không cần đến mô hình kênh thay đổi theo thời gian.

Cáp sợi quang và ống dẫn sóng thuộc loại phương tiện truyền sóng có hướng. Khi các mode truyền lan thay đổi, các kênh này có thể được mô hình hóa là các hệ thống tuyến tính bất biến và được đặc trưng bởi các hàm truyền đạt.

Các hệ thống truyền thông sóng ánh sáng có hướng dùng sợi quang, trong khi đó các hệ thống truyền thông quang không gian tự do truyền ánh sáng qua khí quyển. Hầu hết các hệ thống truyền thông sóng ánh sáng dùng cáp sợi quang đơn mode hoặc đa mode làm kênh truyền, có một nguồn số liệu nhị phân, một máy thu để tạo ra một quyết định dựa trên năng lượng thu trong khoảng thời gian mỗi bit.

Ngoài việc gây suy hao xung truyền dẫn, sợi quang còn gây méo hay dãn rộng các xung phát. Tồn tại hai cơ chế méo: tán sắc và tán sắc liên mode. Tán sắc là hậu quả của sự khác nhau về vận tốc truyền lan của các thành phần phổ được truyền. Tán liên mode được thấy trong sợi đa mode và tổng hợp từ một số lượng lớn các đường truyền lan dọc theo sợi và đến đầu vào bộ tách sóng với các trễ khác nhau. *Đây là hiệu ứng đa đường*. Những mối nối và khớp nối trong một mạng cáp quang gây ra phản xạ mà có thể được xem là tán liên mode phụ. Mô hình kênh đa đường được trình bày sơ bộ ở chương 4, sẽ được nghiên cứu kỹ hơn trong phần 14.4. Phần 14.4 nhấn mạnh kênh vô tuyến pha đinh, *vì vậy nội dung được trình bày ở đây có khả năng áp dụng cho nhiều loại kênh gồm cả cáp kim loại và cáp sợi quang.*

Quan hệ vào/ra của một sợi quang được mô tả bởi hàm truyền đạt tương đương thông thấp:

$$H(f) = \int_{-\infty}^{\infty} S(\lambda).G(f).H_{im}(\lambda).H_c(\lambda, f)d\lambda \tag{14.2}$$

Trong đó phổ nguồn tín hiệu $S(\lambda)$ là một hàm của bước sóng λ, $G(\lambda)$ là độ lợi chọn lọc tần số của sợi, $H_{im}(\lambda)$ là tán liên mode và $H_c(\lambda, f)$ là tán sắc.

Tán liên mode là:

$$H_{im}(f) = \frac{1}{\sigma_{im}\sqrt{2\pi}} e^{\left(-\sigma_{im}^2 (2\pi f)^2 / 2\right) - j2\pi f t_d} \tag{14.3}$$

Trong đó σ_{im} là độ rộng đáp ứng xung **rms** và t_d là trễ thời gian của sợi.

Tán sắc màu là:

$$H_c(\lambda, f) = e^{-j2\pi f \ell T(\lambda)} \tag{14.4}$$

Trong đó ℓ là độ dài sợi quang và $T(\lambda)$ là trễ nhóm của sợi quang.

Tìm được tập các thông số: phổ nguồn $S(\lambda)$, các đặc tính phân tán $T(\lambda)$, suy hao $L(\lambda)$ *từ* nhà sản xuất nguồn quang và sợi quang, được dùng để tính hàm truyền đạt bằng cách thay chúng vào (14.2) và lấy tích phân số tại các tần số f khác nhau. Dùng một số phép lấy xấp xỉ cho $T(\lambda)$ và $S(\lambda)$ để đơn giản quá trình tính toán hàm truyền đạt. Ví dụ, nguồn phổ lý tưởng có thể được coi là xung tần số. Xấp xỉ Gausơ có trung bình λ_0 có thể được dùng cho hầu hết nguồn thực tế. Hàm trễ nhóm thường được lấy xấp xỉ bởi một hàm bậc hai $\lambda - \lambda_0$. Một khi ước lượng tích phân (14.2), kết quả được lưu trữ ở dạng bảng, việc mô phỏng cho kênh này được thực hiện bằng cách sử dụng FIR.

Mô hình được cho ở (14.2) là mô hình hàm truyền đạt vào/ra công suất cho sợi quang, là hợp lý để sử dụng trong các hệ thống truyền thông sóng ánh sáng tách sóng trực tiếp trong đó nguồn phổ rất hẹp so với độ rộng băng thông điều chế. Đối với các hệ thống băng rộng và các hệ thống thông tin quang coherent thì mô hình không hợp lệ.

14.3. Kênh vô tuyến

Các kênh vô tuyến đã được sử dụng từ rất sớm cho các hệ thống truyền thông đường dài, bắt đầu từ thử nghiệm của Marconi trong điện báo vô tuyến. Truyền lan sóng vô tuyến qua khí

quyển gồm cả tầng điện ly - kéo dài vài trăm km trên bề mặt trái đất là một hiện tượng rất phức tạp. Truyền lan qua khí quyển phụ thuộc vào rất nhiều nhân tố như tần số, độ rộng băng tần tín hiệu, kiểu ăng ten, địa thế giữa ăng ten phát và ăng ten thu (nông thôn, thành phố, trong nhà, ngoài trời, v.v...) và các điều kiện thời tiết (khí quyển, mưa, sương mù, v.v...). Các nhà khoa học về khí quyển đã có cố gắng đáng kể để hiểu và triển khai các mô hình nhằm mô tả quá trình truyền lan sóng vô tuyến qua khí quyển. Ngoài ra, rất nhiều chương trình đo đạc đã thu được các dữ liệu truyền dẫn thực nghiệm từ tần số cao HF tới sóng viba. Tất cả những nỗ lực đó đã cho ta cách để mô hình hoá quá trình truyền lan sóng vô tuyến qua khí quyển và cách sử dụng những mô hình đó để phân tích, thiết kế và mô phỏng các hệ thống truyền thông hiện đại. Những tài liệu về mô hình hoá kênh vô tuyến rất phong phú và mọi nỗ lực để tóm tắt chúng trong vài trang giấy là không đủ. Tuy nhiên, cố gắng trình bầy một vài giải pháp để mô hình hoá và mô phỏng các hệ thống truyền thông.

Từ quan điểm thiết kế hệ truyền thông, ta phân các mô hình truyền lan thành hai loại sau: *(i) mô hình ảnh hưởng phạm vi rộng* (được dùng để tính toán tổn hao truyền sóng); *(ii) mô hình ảnh hưởng phạm vi hẹp* (được dùng để mô hình hoá méo thông tin do hiệu ứng đa đường, hoặc do sự thay đổi ngẫu nhiên đặc tính truyền lan của kênh). Trong khi, loại 1 được dùng để thiết lập độ dự trữ công suất tuyến và phân tích vùng phủ sóng trong giai đoạn thiết kế khởi đầu, thì loại thứ hai được dùng để thiết kế chi tiết cho hệ thống. Ta sẽ tập trung vào loại mô hình thứ hai với mục đích nhấn mạnh các giải pháp để mô phỏng hiệu quả.

Ta bắt đầu bằng các mô hình kênh với một kênh "gần như" là không gian tự do để khảo sát vùng truyền lan sóng giữa ăng ten phát/thu là hoàn toàn tự do (không xảy ra các hiện tượng hấp thụ hoặc phản xạ sóng vô tuyến RF). Khi này, coi khí quyển là môi trường hoàn toàn đồng nhất, không hấp thụ.

Trong mô hình được lý tưởng hóa này, kênh chỉ làm suy hao tín hiệu và không gây méo dạng sóng. Suy hao được tính theo *mô hình truyền lan không gian tự do* được định nghĩa là:

$$L_f = \left(\frac{4\pi d}{\lambda}\right)^2 \tag{14.5}$$

Trong đó λ là bước sóng của tín hiệu phát, d là khoảng cách ăng ten phát/thu vô hướng. Khuếch đại ăng ten phát và ăng ten thu được dùng để tính toán công suất thu thực tế.

Vì đa số các kênh thực tế trong đó tín hiệu truyền lan qua khí quyển và gần mặt đất, nên giả định kênh truyền lan không gian tự do không còn phù hợp nữa. Ảnh hưởng đầu tiên phải xét đến là khí quyển, nó gây ra hấp thụ, khúc xạ và tán xạ. Với băng hẹp, hấp thụ do khí quyển dẫn đến suy hao phụ. Tuy nhiên, với băng rộng tổn hao lại phụ thuộc vào tần số và thường được mô hình hóa bởi một hàm truyền đạt. Hiệu ứng *lọc* này được xét bất biến theo thời gian hoặc ít nhất là *tựa tĩnh* (vì kênh thay đổi rất chậm sơ với tín hiệu). Một hiện tượng thuộc khí quyển khác nữa là méo pha do tầng điện ly, có thể được mô hình hóa bởi một đáp ứng pha thay đổi chậm theo thời gian hoặc bất biến theo thời gian. Một số ví dụ về các mô hình hàm truyền đạt được dùng để đặc tính hóa các loại kênh khí quyển cụ thể được mô tả dưới đây.

Các ảnh hưởng của khí quyển khác, mặt đất, những vật thể có trên mặt đất gần với đường truyền sóng thường tạo ra truyền dẫn đa đường. *Truyền dẫn đa đường là tín hiệu đến qua nhiều*

đường phản xạ và/hoặc khúc xạ từ máy phát đến máy thu. Các ảnh hưởng này có thể thay đổi theo thời gian (do sự thay đổi trong các điều kiện khí quyển hoặc sự chuyển động tương đối của ăng ten phát và ăng ten thu như trong thông tin di động). Thuật ngữ *tia lửa* được bắt nguồn từ thiên văn học vô tuyến, được sử dụng để mô tả tính thay đổi theo thời gian trong các đặc tính kênh *do* các thay đổi về vật lý trong môi trường truyền lan như các thay đổi mật độ điện tử trong tầng điện ly gây phản xạ sóng vô tuyến tần số cao. Thuật ngữ pha đinh đa đường được dùng trong truyền thông di động để mô tả các thay đổi trong các điều kiện kênh và trong các đặc tính tín hiệu thu. Phần cuối của chương sẽ xét các mô hình kênh pha đinh đa đường.

14.3.1. Kênh tầng đối lưu

Truyền thông tầng đối lưu (không phải điện ly) sử dụng băng tần VHF (30 đến 300 MHz) và UHF (300 MHz tới 3 GHz) để truyền tin trên cự ly vài nghìn km. Trong những băng tần này, oxy và hơi nước có trong khí quyển sẽ hấp thụ năng lượng RF. Tổn hao do hấp thụ phụ thuộc vào tần số của sóng RF cũng như điều kiện khí quyển, độ ẩm thực tế. Tập các đặc tính điển hình về tổn hao truyền lan do sự hấp thụ khí quyển thường được thấy ở nhiều tài liệu.

Các đặc tính hấp thụ mang tính chọn lọc tần số của khí quyển có thể được xấp xỉ bằng một hàm truyền đạt ở dạng:

$$H(f) = H_0 e^{j0{,}02096 f\left[10^6 + N(f)\ell\right]} \tag{14.6}$$

Trong đó $N(f)$ là khúc xạ phức của khí quyển và được cho bởi:

$$N(f) = N_0 + D(f) + jN^{*}(f) \tag{14.7}$$

Trong đó các biểu thức (14.6) và (14.7) thì H_0 là hằng số, N_0 là khúc xạ phụ thuộc tần số, $D(f)$ là hấp thụ khúc xạ, $N^{*}(f)$ là sự hấp thụ, ℓ là khoảng cách (km). Các tham số này phụ thuộc vào tần số và các điều kiện khí quyển như nhiệt độ, áp suất, độ ẩm. Các giá trị điển hình thường được trình bày ở dạng bảng dữ liệu.

Một khi biết trước các điều kiện khí quyển và độ rộng băng tần của tín hiệu, thì có thể tính được hàm truyền đạt theo thực nghiệm tại các giá trị tần số khác nhau theo (14.6) và (14.7). Có thể tìm được hàm truyền đạt tương đương thông thấp bằng cách dịch tần số và dùng kỹ thuật FIR để mô phỏng mô hình kênh này.

14.3.2. Ảnh hưởng của mưa lên kênh vô tuyến

Mưa có ảnh hưởng đáng kể lên truyền lan sóng viba tại các tần số cao (lớn hơn 10 GHz). Nhiều kỹ thuật đã được đề xuất trong các tài liệu để mô hình hoá các ảnh hưởng của mưa. Suy hao do mưa rào là một hàm của tốc độ mưa và tần số. Khi tốc độ mưa, tần số tín hiệu càng lớn *thì* suy hao càng lớn. Theo đó, suy hao tăng khi tốc độ mưa và tần số tăng. Ngoài ra, xuất hiện các đỉnh cộng hưởng trong đặc tính suy hao. Các đỉnh cộng hưởng thường xảy ra tại 22 GHz và 60 GHz. Đỉnh tại 22 GHz xảy ra do hơi nước và đỉnh tại 60 GHz xảy ra do các phân tử oxy.

Suy hao do mưa rào thường được tính toán cho các vị trí địa lý cụ thể sử dụng tính thống kê về tỉ lệ mưa vùng đó. Với thông tin vệ tinh, tổn hao được tính toán là một hàm của góc ngẩng ăng ten trạm mặt đất (theo chiều ngang) và tần số. Độ ngẩng càng thấp có nghĩa là có

nhiều nước mưa hơn và tổn hao càng cao. Ảnh hưởng của mưa rào được tính toán cụ thể với xác suất ngừng hoạt động cho trước, là phân số của thời gian mà BER của tuyến vượt giá trị ngưỡng (thường là 10^{-3} cho thông tin thoại và 10^{-6} cho các tuyến số liệu).

Với các độ rộng băng thông (dải thông) tương đối nhỏ, các ảnh hưởng của mưa có thể được tính đến bằng cách đưa thêm thành phần tổn hao phụ vào mô hình kênh. Tuy nhiên, khi dải thông của tín hiệu lớn, sẽ thay đổi suy hao trên băng thông và cần có một mô hình kiểu hàm truyền đạt. Đáp ứng biên độ của hàm truyền đạt có một độ nghiêng tuyến tính (trên độ chia theo log (dB)) và đáp ứng pha được coi là tuyến tính.

Các kênh tryền lan không gian tự do ở tần số cao sử dụng ăng ten tính hướng cao, có đặc tính phân cực cụ thể. Khi bước sóng (tương ứng tần số sóng mang) lớn hơn rất nhiều so với kích thước phần tử khí quyển và khi không có các vật cản vật lý đối với đa đường, thì có thể sử dụng phân cực ăng ten để cách ly (phân lập) các kênh. Trong các hệ thống truyền thông sử dụng nhiều phân cực trực giao cho các tín hiệu khác nhau, thì phải xét đến hiệu ứng khử cực của mưa. Khử phân cực có nghĩa là năng lượng trong một phân cực rò ra hoặc kết hợp (kết hợp với năng lượng trong sự phân cực trực giao). Điều này gây ra xuyên âm.

Nếu có hai tín hiệu được truyền trên phân cực trực giao là:

$$\tilde{s}_i(t) = A_i(t)e^{j\phi_i(t)}, \quad i = 1,2 \tag{14.8}$$

Thì mô hình đơn giản nhất cho hai tín hiệu thu với khử phân cực là:

$$\begin{aligned} \tilde{r}_1(t) &= \alpha_{11}\tilde{s}_1(t) + \alpha_{12}\tilde{s}_2(t) \\ \tilde{r}_2(t) &= \alpha_{21}\tilde{s}_1(t) + \alpha_{22}\tilde{s}_2(t) \end{aligned} \tag{14.9}$$

Trong đó tỉ số $20\log\left(\alpha_{11}/\alpha_{21}\right)$ là số đo nhiễu phân cực chéo (XPI) lên tín hiệu 1 từ tín hiệu 2. Trong khi rất nhiều phép lấy xấp xỉ khả dụng trong tài liệu để phân tích ảnh hưởng của XPI lên hệ thống số hoặc tương tự, thì cần thiết phải tăng cường mô phỏng khi hệ thống lệch khỏi hệ thống lý tưởng.

14.4. Kênh pha đinh đa đường

14.4.1. Giới thiệu

Ta mô hình hóa và mô phỏng hiện tượng đa đường và tính chuyển động được quy vào pha đinh, là hai hiện tượng ảnh hưởng nghiêm trọng nhất lên hiệu năng. Bất kỳ kênh truyền thông vô tuyến nào cũng có nhiều hơn một đường truyền giữa ăng ten phát và ăng ten thu. Truyền sóng đa đường có thể dọ phản xạ hoặc khúc xạ khí quyển hoặc do phản xạ từ các toà nhà hay các vật thể khác. Pha đinh đa đường có thể xảy ra ở tất cả các hệ thống truyền thông vô tuyến. Các hiệu ứng này lần đầu tiên được quan sát và phân tích cho các hệ thống tán xạ đối lưu HF những năm 1950 và 1960. Các quan tâm hiện nay là vấn đề mô hình hóa và mô phỏng cho pha đinh đa đường trong hệ thống vô tuyến trong nhà và di động ở dải tần số 1—60 GHz. Mặc dù cơ chế pha đinh là khác nhau nhưng các khái niệm về mô hình hóa, phân tích, mô phỏng là giống nhau.

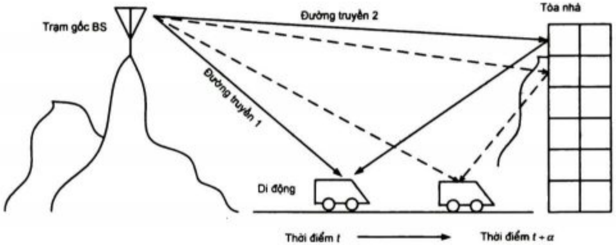
14.4.2. Kênh pha đinh đa đường điển hình

Để minh hoạ giải pháp cơ bản cho việc lập mô hình kênh pha đinh, ta xét một kênh truyền thông di động có hai đường (tia) riêng biệt từ máy di động tới một trạm gốc cố định

được minh hoạ ở hình 14.1a. Mặc dù hình 14.1a có hai đường nhưng dễ dàng tổng quát hoá cho N đường. Trường hợp N đường đầu ra kênh (tín hiệu vào máy thu di động) là:

$$y(t)=\sum_{n=1}^{N}a_n(t)x\big(t-\tau_n(t)\big) \tag{14.10}$$

Trong đó $a_n(t)$, $\tau_n(t)$ là suy hao và trễ truyền dẫn của thành phần đa đường thứ n. Lưu ý rằng, trễ và suy hao là hàm của thời gian để chỉ ra rằng khi ô tô di chuyển, thì trễ và suy hao cũng như nhiều thành phần đa đường thay đổi là một hàm của thời gian. Tín hiệu tổng tại máy thu là tổng của một lượng lớn các thành phần tán xạ. Những thành phần này cùng với các pha ngẫu nhiên và do đó đường bao phức kết quả có thể được mô hình hoá là một quá trình Gausơ *phức* theo định lý giới hạn trung tâm. Sự di động trong khoảng cách nhỏ là bội của $\lambda/2$ (khoảng 15 cm tại 1 GHz) làm thay đổi pha đáng kể của các thành phần tán xạ trong một tia, và làm cho các thành phần đó cộng với nhau tại một vị trí và loại trừ nhau ở vị trí lân cận. Điều này dẫn đến công suất/biên độ tín hiệu thu thay đổi nhanh được gọi là *pha đinh nhanh hay pha đinh phạm vi hẹp*.



Hình 14.1a: Minh họa kênh pha đinh đa đường

Cần lưu ý rằng, pha đinh phạm vi hẹp do các thay đổi pha gây ra chứ không phải là do suy hao, vì thay đổi các độ dài đường truyền một khoảng nhỏ trên các khoảng cách nhỏ. Mặt khác, nếu MS di chuyển trên khoảng cách lớn và chiều dài đường truyền tăng từ 1 lên 2 km, thì cường độ tín hiệu thu sẽ bị suy giảm vì suy hao thay đổi đáng kể. Tính di chuyển trên khoảng cách lớn ($>>\lambda$) và sự thay đổi đặc điểm địa hình, sẽ ảnh hưởng tới suy hao và công suất thu thay đổi chậm. Hiện tượng này được gọi là *pha đinh chậm* hay *pha đinh phạm vi rộng*, và được mô hình hóa riêng biệt được thảo luận trong phần tiếp theo của chương này.

Ta đã thấy đường bao phức của đầu vào máy thu từ số lượng lớn các thành phần tán xạ là một quá trình Gausơ phức. Khi này, quá tình ngẫu nhiên có giá trị trung bình không và độ lớn của quá trình là phân bố Rayleigh. Nếu tồn tại thành phần đi thẳng LOS thì quá trình trở thành quá trình Ricean. Ảnh hưởng của quá trình này sẽ được giải thích trong ví dụ 14.1.

Trong phạm vi nhất định, định nghĩa về pha đinh nhanh và chậm có thể nhận thấy được. Tuy nhiên, khi nói về pha đinh nhanh và chậm, ta thường nghĩ về tốc độ ký hiệu cơ bản. Cụ thể, kênh pha đinh chậm được định nghĩa là kênh trong đó mức tín hiệu thu về cơ bản là không đổi trong nhiều ký hiệu hoặc khung dữ liệu. Pha đinh nhanh có nghĩa là cường độ trường tín hiệu

thu thay đổi đáng kể trong các khoảng thời gian ký hiệu. Vì thế, định nghĩa về pha đinh nhanh và chậm phụ thuộc vào tốc độ ký hiệu.

Tại đây, ta xác định đường bao phức của tín hiệu thu.

Giả sử đầu vào kênh truyền sóng là tín hiệu điều chế dạng:

$$x(t) = A(t).\cos\left(2\pi f_c t + \phi(t)\right) \tag{14.11}$$

Vì thực hiện mô phỏng dạng sóng bằng cách sử dụng các tín hiệu đường bao phức, nên ta phải xác định đường bao phức cho cả x(t) và y(t) là $\tilde{x}(t)$ và $\tilde{y}(t)$ từ đó rút ra $\tilde{h}(t,\tau)$.

Đường bao phức của tín hiệu phát $\tilde{x}(t)$: Bằng cách kiểm tra (14.11) ta có:

$$\tilde{x}(t) = A(t).e^{\phi(t)} \tag{14.12}$$

Đường bao phức của tín hiệu thu $\tilde{y}(t)$: Được xác định như sau

Thay (14.11) vào (14.10) ta có tín hiệu thu *y(t)*:

$$y(t) = \sum_{n=1}^{N} a_n(t).A\left(t-\tau_n(t)\right).\cos\left[2\pi f_c\left(t-\tau_n(t)\right)+\phi\left(t-\tau_n(t)\right)\right] \tag{14.13}$$

Có thể viết là:

$$y(t) = \sum_{n=1}^{N} a_n(t).A\left(t-\tau_n(t)\right).\mathrm{Re}\left\{e^{j\phi(t-\tau_n(t))}\, e^{-j2\pi f_c \tau_n(t)}\, e^{j2\pi f_c t}\right\} \tag{14.14}$$

Vì $a_n(t)$ và $A(t)$ đều là giá trị thực, nên (14.4) được viết là:

$$y(t) = \mathrm{Re}\left\{\sum_{n=1}^{N} a_n(t).A\left(t-\tau_n(t)\right).e^{j\phi(t-\tau_n(t))}\, e^{-j2\pi f_c \tau_n(t)}\, e^{j2\pi f_c t}\right\} \tag{14.15}$$

Từ (14.12) ta có:

$$A\left(t-\tau_n(t)\right).e^{j\phi(t-\tau_n(t))} = \tilde{x}\left(t-\tau_n(t)\right) \tag{14.16}$$

Vì thế:

$$y(t) = \mathrm{Re}\left\{\sum_{n=1}^{N} \underbrace{a_n(t).e^{-j2\pi f_c \tau_n(t)}}_{\tilde{a}_n(t)}\, \tilde{x}\left(t-\tau_n(t)\right).e^{j2\pi f_c t}\right\} \tag{14.17}$$

Suy hao đường truyền phức được định nghĩa là:

$$\tilde{a}_n(t) = a_n(t).e^{-j2\pi f_c \tau_n(t)} \tag{14.18}$$

Vì vậy:

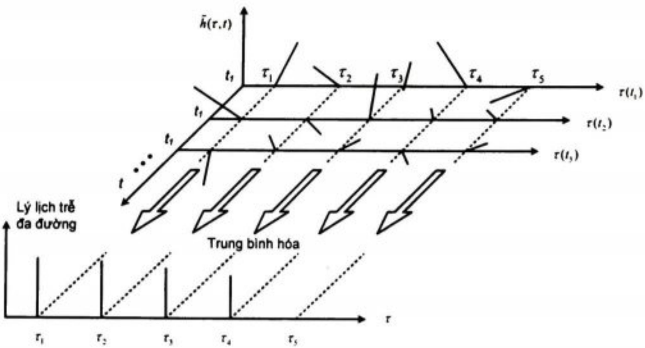
$$y(t) = \mathrm{Re}\left\{\sum_{n=1}^{N} \tilde{a}_n(t).\tilde{x}\left(t-\tau_n(t)\right).e^{j2\pi f_c t}\right\} \tag{14.19}$$

Vì vậy, đường bao phức của tín hiệu thu *y(t)* là:

$$\tilde{y}(t) = \sum_{n=1}^{N} \tilde{a}_n(t).\tilde{x}\left(t-\tau_n(t)\right) \tag{14.20}$$

Rút ra đáp ứng xung kim của kênh $\tilde{h}(\tau,t)$: Quan hệ vào/ra của kênh được định nghĩa bởi (14.20) tương ứng một hệ thống tuyến tính thay đổi theo thời gian LTV có đáp ứng xung kim là:

$$\tilde{h}(\tau,t)=\sum_{n=1}^{N}\tilde{a}_n(t).\delta\left(t-\tau_n(t)\right) \tag{14.21}$$



Hình 14.1b: Minh họa đáp ứng xung kim của kênh và lý lịch trễ đa đường

Trong (14.21), $\tilde{h}(\tau,t)$ là đáp ứng xung của kênh đo được tại thời điểm t với giả thiết rằng xung kim được đặt vào kênh tại thời điểm $t-\tau$. Theo đó, τ trình bày cho thời gian trôi qua hoặc trễ truyền lan. Hình 14.1b minh họa đáp ứng xung kim của kênh $\tilde{h}(\tau,t)$ và lý lịch trễ đa đường.

Trường hợp đặc biệt: Trường hợp không di chuyển hoặc không có sự thay đổi trong môi trường truyền dẫn, thì quan hệ vào/ra là bất biến theo thời gian mặc dù đa đường vẫn xuất hiện. Khi này, trễ truyền dẫn tương ứng với đường truyền thứ n và suy hao đường truyền đó là không đổi $\tilde{a}_n(t)=\tilde{a}_n$ & $\tau_n(t)=\tau_n$ (kênh cố định):

$$\tilde{y}(t)=\sum_{n=1}^{N}\tilde{a}_n.\tilde{x}\left(t-\tau_n\right) \tag{14.22}$$

Biểu diễn kênh trong các miền thời gian và tần số như sau:

$$\tilde{h}(\tau)=\sum_{n=1}^{N}\tilde{a}_n.\delta\left(t-\tau_n\right) \tag{14.23}$$

$$H(f)=\sum_{n=1}^{N}\tilde{a}_n.e^{-j2\pi f\tau_n} \tag{14.24}$$

Thấy rõ, trường hợp kênh bất biến theo thời gian LTIV, kênh chỉ thực hiện như một bộ lọc đối với tín hiệu phát.

Ví dụ 14.1: Ta mô phỏng hiệu năng BER hệ thống QPSK hoạt động trong môi trường kênh đa đường 3 tia cố định với AWGN và so sánh hiệu năng BER với chính hệ thống đó

nhưng hoạt động trên kênh AWGN lý tưởng (không có đa đường). Để đơn giản mô hình mô phỏng, ta giả sử như sau:

1. Kênh có 3 đường (tia) gồm có: 1 đường đi thẳng LOS không bị pha đinh và 2 thành phần Rayleigh. Các mức công suất thu liên quan tới mỗi đường và các trễ *chênh lệch* giữa ba đường là các tham số mô phỏng.

2. Pha đinh Rayleigh trong kênh chỉ ảnh hưởng lên *biên độ* của tín hiệu phát. Pha tức thời không bị ảnh hưởng.

3. Suy hao của mỗi thành phần đa đường là không đổi trên mỗi khoảng thời gian ký hiệu và có các giá trị độc lập trên các khoảng liền kề (không yêu cầu phải định dạng phổ Doppler).

4. Không sử dụng bộ lọc phát và mô hình máy thu là máy thu kết xuất và tích hợp lý tưởng.

Với các giả định trên thì, tín hiệu thu cho ví dụ này được viết là:

$$\tilde{y}(t) = \underbrace{a_0\tilde{x}(t)}_{\text{LOS}} + \underbrace{a_1R_1\tilde{x}(t)}_{\text{Rayleigh}} + \underbrace{a_2R_2\tilde{x}(t-\tau)}_{\text{Rayleigh bị trễ}} \tag{14.25}$$

Trong đó R_1 và R_2 là hai biến ngẫu nhiên Rayleigh độc lập thể hiện cho suy hao của hai đường Rayleigh và τ là trễ *tương đối* giữa hai thành phần Rayleigh. Biến đổi Fourier của (14.25) là:

$$\tilde{Y}(f) = a_0\tilde{X}(f) + a_1R_1\tilde{X}(f) + a_2R_2\tilde{X}(f)e^{-j2\pi f\tau} \tag{14.26}$$

Dẫn đến hàm truyền đạt kênh là:

$$\tilde{H}(f) = \frac{Y(f)}{X(f)} = a_0 + a_1R_1 + a_2R_2\,e^{-j2\pi f\tau} \tag{14.27}$$

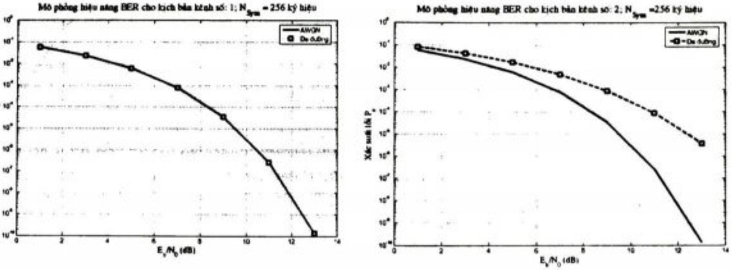
Rõ ràng, nếu tích $f\tau$ không thể bỏ qua so với độ rộng băng tần của tín hiệu, thì kênh là kênh chọn lọc tần số, dẫn đến trải rộng trễ và ISI. Các giá trị của a_0, a_1, a_2 xác định các mức công suất tương đối P_0, P_1 và P_2 của ba thành phần đa đường.

Mô phỏng được thực hiện cho một trong sáu tập giá trị tham số được cho ở bảng 14.1. Với mỗi kịch bản kênh, BER được ước lượng dùng ước tính bán phân tích. Trong bảng 14.1, trễ được biểu diễn theo chu kỳ lấy mẫu. Vì tần số lấy mẫu mô phỏng là 16 mẫu trên ký hiệu, nên $\tau = 8$ tương đương với trễ là một nửa chu kỳ lấy mẫu. (Mã chương trình Matlab được cho ở **NVD14_threeray.m** trong Phụ lục 14A)

Bảng 14.1: Các kịch bản kênh để minh họa cho pha đinh

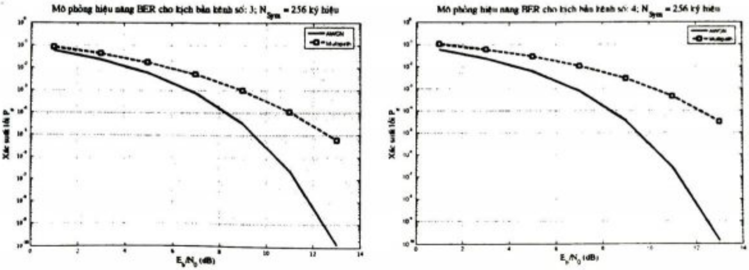
Kịch bản kênh số	P_0	P_1	P_2	τ (mẫu)	Chú thích
1	1,0	0	0	0	Dùng để phê chuẩn (kênh lý tưởng AWGN)
2	1,0	0,2	0	0	Pha đinh phẳng Ricean
3	1,0	0	0,2	0	Pha đinh phẳng Ricean
4	1,0	0	0,2	8	Pha đinh chọn lọc tần số Ricean
5	0	1,0	0,2	0	Pha đinh phẳngRayleigh
6	0	1,0	0,2	8	Pha đinh chọn lọc tần số Rayleigh

Kết quả mô phỏng đối với kịch bản kênh số 1 và 2 được minh hoạ ở hình 14.2. Trong kịch bản số 1, chỉ có một thành phần đi thẳng LOS mà không có đa đường, dẫn đến ước tính BER bán phân tích cho hệ thống QPSK hoạt động trong môi trường kênh AWGN. Việc mô phỏng này nhằm để kiểm tra phương pháp luận mô phỏng và tạo ra các *kết quả ranh giới* biểu diễn cho một hệ thống QPSK lý tưởng. Vì mục đích so sánh, nên kết quả này được hiển thị cùng với kết quả BER của 5 kịch bản kênh còn lại. Bảng 14.1 cho thấy, kịch bản 2 là kịch bản số 1 thêm một thành phần pha đinh Rayleigh. Việc thêm vào này có thể đáp ứng với kênh pha đinh Ricean. Vì $\tau = 0$, nên kịch bản kênh số 2 là pha đinh phẳng (không chọn lọc tần số). Thấy rõ, BER tăng lên so với kịch bản kênh số 1 (trường hợp kênh lý tưởng).



Hình 14.2: Kịch bản kênh số 1 (trái) và kịch bản kênh số 2 (phải) minh hoạ chạy hiệu chuẩn và pha đinh phẳng Ricean.

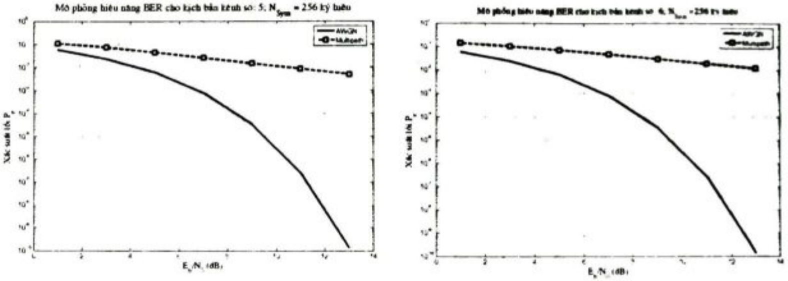
Kết quả mô phỏng cho kịch bản kênh số 3 và 4 được minh hoạ ở hình 14.3. Kịch bản kênh số 3 về cơ bản tương đương kịch bản kênh số 2. Sự khác nhau nhỏ là do quá trình pha đinh khác với quá trình pha đinh được dùng ở kịch bản kênh số 2 do sự khởi tạo khác nhau của bộ tạo số ngẫu nhiên cơ bản. Kịch bản kênh số 4 giống với kịch bản kênh số 3 ngoại trừ là pha đinh bây giờ là chọn lọc tần số. Lưu ý hiệu năng hệ thống bị suy giảm hơn.



Hình 14.3: Lớp 3 (bảng bên trái) và lớp 4 (bảng bên phải) minh hoạ pha đinh nền Ricean và pha đinh chọn lọc tần số

Kết quả mô phỏng cho kịch bản kênh số 5 và 6 được cho ở hình 14.4. Lưu ý rằng, cả hai kịch bản kênh này đều không có thành phần đi thẳng. Sự so sánh kết quả kịch bản số 5 với 4

kết quả trước cho thấy: kể cả với kịch bản pha đinh phẳng (bên trái), hiệu năng đều xấu hơn so với trường hợp có thành đi thẳng LOS. Kịch bản kênh số 6 tương tự với kịch bản kênh số 5 ngoại trừ là pha đinh bây giờ là chọn lọc tần số. Lưu ý rằng hiệu năng hệ thống bị suy thoái hơn. Các kênh Ricean và Rayleigh sẽ được khai thác sâu hơn trong các phần tiếp theo.



Hình 14.4: Kịch bản kênh số 5 (trái) và số 6 (phải) minh hoạ pha đinh chọn lọc tần số và pha đinh phẳng Rayleigh

14.4.3. Đa đường khuếch tán và đa đường rời rạc

Nhiều thành phần đa đường sẽ thay đổi phụ thuộc vào loại kênh. Trong các tuyến truyền thông viba sử dụng ăng ten tính hướng cao (búp sóng hẹp), số lượng các thành phần đa đường sẽ nhỏ, ngược lại ở hệ thống thông tin di động vùng thành phố sử dụng các ăng ten vô hướng do phản xạ từ các toà nhà, nên tạo ra rất nhiều thành phần đa đường. Tương tự xảy ra đối với truyền thông tin vô tuyến trong nhà, ở đó các tín hiệu có thể bung ra khỏi tường, nội thất và các bề mặt khác.

Tồn tại một số trạng thái giống như các kênh tán xạ tầng đối lưu, hoặc các kênh vô tuyến di động, ở đó phù hợp để nhìn nhận tín hiệu thu gồm sự liên tục các thành phần đa đường chứ không phải là tập hợp các thành phần rời rạc. Trạng thái này được gọi *là đa đường khuếch tán*. Ta sẽ thấy rằng: kênh đa đường khuếch tán có thể được xấp xỉ bằng (phiên bản được lấy mẫu của) kênh đa đường rời rạc cho mục đích mô phỏng.

14.5. Mô hình hoá kênh pha đinh đa đường

Tài liệu gần đây về các hệ thống truyền thông gồm rất nhiều bài báo đề cập vấn đề mô hình hoá và phân tích kênh pha đinh đa đường, đặc biệt là các kênh di động ngoài trời và vô tuyến trong nhà. Trong khi việc xét lại đầy đủ là ngoài phạm vi của chương này, nhưng ta cũng xét ngắn gọn về mô hình hoá kênh vô tuyến di động ngoài trời để triển khai các kỹ thuật mô phỏng. Các kỹ thuật mô hình hóa và mô phỏng này có thể áp dụng cho các kênh pha đinh đa đường khác.

Mô hình hoá một kênh di động ngoài trời thường được tiến hành hai bước, nó biểu diễn các ảnh hưởng phạm vi hẹp và phạm vi rộng của pha đinh và đa đường. Như đã đề cập, pha đinh phạm vi rộng thể hiện cho suy hao hoặc tổn hao đường truyền trên một vùng rộng và hiện tượng này do sự nhô lên của các đặc điểm địa hình như đồi núi, toà nhà, v.v... giữa máy phát và máy thu. Máy thu thường bị ẩn hoặc bị che chắn do các đặc điểm địa hình, tính thống kê của

pha đinh phạm vi rộng tạo ra cách thức tính toán công suất tín hiệu thu hay tổn hao đường truyền *là hàm của khoảng cách*. Pha đinh phạm vi hẹp liên quan với tính *thay đổi động* về biên độ và pha của tín hiệu thu, là kết quả của các thay đổi rất nhỏ trong phân tách không gian giữa máy phát và máy thu.

Ba cơ chế ảnh hưởng lên chất lượng tín hiệu thu trong một kênh di động: *phản xạ, khúc xạ và tán xạ*. *Phản xạ* xảy ra khi sóng vô tuyến đụng phải một bề mặt lớn và phẳng (như bề mặt nước hoặc bề mặt kim loại lớn). *Nhiễu xạ* xẩy ra tại nơi có sự cản trở trên đường vô tuyến giữa máy phát và máy thu gây ra một sóng vô tuyến thứ cấp tạo ra sự cản trở sau đó. Điều này được gọi là che chắn (*shadowing*) và giải thích cho sóng vô tuyến vẫn đến được ăng ten thu mặc dù không tồn tại đường nhìn thẳng LOS giữa máy phát và máy thu. *Tán xạ*: kết quả từ các bề mặt gồ nghề mà có kích thước là bội số của bước sóng, làm cho năng lượng bị phản xạ đó tán xạ ra mọi hướng.

Trong khi lý thuyết trường điện từ đưa ra các mô hình rất phức tạp cho những hiện tượng này, thì ta có thể sử dụng các mô hình thống kê đơn giản hơn cho quan hệ vào/ra cho kênh di động. Cụ thể, có thể mô hình hóa đáp ứng tương đương thông thấp của một kênh vô tuyến di động bởi đáp ứng xung kim phức dạng:

$$\tilde{h}(\tau,t)=\underbrace{\left\{\left[\frac{k}{d^n}g_{sh}\left(p(t)\right)\right]^{1/2}\right\}}_{\text{Pha đinh phạm vi rộng}}.\ \underbrace{\tilde{c}\left(\tau,p(t)\right)}_{\text{Pha đinh phạm vi hẹp}}\ ,\quad d>1\,km \tag{14.28}$$

Trong đó thành phần trong ngoặc kép mô hình hóa cho pha đinh phạm vi rộng, $\tilde{c}(\tau,p(t))$ giải thích cho pha đinh phạm vi hẹp là một hàm của vị trí của $p(t)$ tại thời điểm t. Hằng số $K = -10\ lg(k)$ là tổn hao trung điểm dB tại khoảng cách 1 km. Vì khoảng cách tham chiếu là 1 km, nên (14.28) chỉ hợp lệ khi d > 1 km. Điển hình, K là bậc của 87 dB tại 900 MHz, d là khoảng cách giữa máy phát và máy thu (đơn vị m) và số mũ tổn hao đường truyền n có giá trị là 2 đối với không gian tự do (hầu hết các kênh di động có giá trị trong khoảng 2 đến 4, các giá trị lớn hơn áp dụng cho các đường truyền bị che khuất). Thừa số $g_{sh}(p(t)$ giải thích cho hiện tượng che chắn do các toà nhà, những che khuất khác tại vị trí $p(t)$ cho trước, và $G=10\log_{10}\left[g_{sh}\left(p(t)\right)\right]$ thường được mô hình hóa là một biến Gausơ có trung bình 0 dB, và độ lệch chuẩn từ 6 đến 12 dB phụ thuộc vào môi trường (mô hình này được gọi là mô hình che chắn loga). Thực tế thường biểu diễn tổn hao đường truyền [số hạng trong dấu ngoặc móc trong (14.28)] là:

$$L(d)_{dB}=L(1km)_{dB}+10.n.\log(d)+X_\sigma \tag{14.29}$$

Trong đó X_σ là biến Gausơ trung bình không với độ lệch chuẩn từ 6 đến 12 dB.

Trong biểu thức (14.28), $\tilde{c}(\tau,p(t))$ trình bày đáp ứng xung kim tương đương thông thấp phức của kênh tại vị trí $p(t)$, đa đường địa phương và pha đinh do dịch chuyển không gian nhỏ xung quanh vị trí định vị $p(t)$. Tổn hao truyền sóng liên quan với pha đinh phạm vi rộng được biểu diễn bởi thành phần trong dấu ngoặc ở (14.28) như pha đinh do che chắn, các thay đổi rất chậm là một hàm của thời gian tại tốc độ di chuyển chuẩn hóa so với tốc độ thay đổi của

$\tilde{c}(\tau, p(t))$. Vì vậy, suy hao kênh do pha đinh phạm vi rộng và che chắn có thể được xem là không đổi trong một vùng nhỏ, và ảnh hưởng phạm vi rộng lên hiệu năng hệ thống được phản ánh trong tín hiệu thu *trung bình*. Tính cách động của các phân hệ máy thu như các vòng bám và các bộ cân bằng, cũng như tỉ số lỗi bit trong hệ thống sẽ bị ảnh hưởng nhiều bởi tính cách phạm vi hẹp mà được mô hình hóa bởi $\tilde{c}(\tau, p(t))$. Vì vậy nhiều cố gắng nỗ lực trong việc lập mô hình và mô phỏng các kênh vô tuyến di động tập trung vào $\tilde{c}(\tau, p(t))$. Dưới đây, ta sử dụng ký hiệu $\tilde{c}(\tau, t)$ thay cho $\tilde{c}(\tau, p(t))$.

14.6. Mô hình quá trình ngẫu nhiên

Rất nhiều mô hình được đề xuất để đặc tính hóa các kênh pha đinh đa đường, và hầu hết đều dùng các mô hình quá trình ngẫu nhiên để đặc tính hóa pha đinh. Hai lớp mô hình để mô tả đa đường: *(i)* mô hình đa đường rời rạc (hữu hạn các thành phần đa đường); *(ii)* mô hình đa đường khuếch tán (tính liên tục các thành phần đa đường). Trong truyền thông vô tuyến di động, mô hình đầu tiên thường được dùng để mô phỏng mức dạng sóng cho các kênh vô tuyến di động, trong khi đó mô hình thứ hai được dùng cho các kênh tán xạ tầng đối lưu với điều chế băng hẹp. Trong cả hai trường hợp này, kênh được mô hình là một hệ thống tuyến tính thay đổi theo thời gian LTV, đầu ra của kênh gồm tổng của N phiên bản đầu vào bị trễ và suy hao. Vì vậy:

$$\tilde{y}(t) = \sum_{k=1}^{N(t)} \tilde{a}_k(t).\tilde{x}\left(t - \tau_k(t)\right) \tag{14.30}$$

Đáp ứng xung kim $\tilde{c}(\tau, t)$ là:

$$\tilde{c}(\tau, t) = \sum_{k=1}^{N(t)} \tilde{a}_k(t).\delta\left(\tau - \tau_k(t)\right) \tag{14.31}$$

Trong đó *N(t)* là số lượng thành phần đa đường và $\tilde{a}_k(t)$ và $\tau_k(t)$ là suy hao phức và trễ của đa đường thứ *k* tại thời điểm *t*.

Như đã đề cập ở trên, một kênh đa đường có thể không thay đổi theo thời gian. Tuy nhiên, tất cả các kênh thực tế đều được đặc tính hóa là thay đổi theo thời gian. Tính thay đổi theo thời gian vì hai lý do

1. Môi trường thay đổi mặc dù máy phát và máy thu cố định; chẳng hạn sự thay đổi của tầng điện ly, sự di chuyển của các bộ tán xạ và phản xạ...

2. Tính di động của máy phát và máy thu mặc dù môi trường cố định. Do đó, ở các kênh pha đinh thực tế thì N, a_k, *và* τ_k *đều có thể thay đổi ngẫu nhiên theo thời gian. Ví dụ minh họa được cho ở hình 14.5a.*

Tín hiệu thu thay đổi ngẫu nhiên do pha đinh có thể được mô hình bằng cách coi $\tilde{c}(\tau, t)$ là một quá trình ngẫu nhiên theo *t*. Nếu tín hiệu thu là tổng của một số lượng lớn các thành phần tán xạ trong mỗi đường truyền, thì theo định lý giới hạn trung tâm dẫn đến một mô hình trong đó $\tilde{c}(\tau, t)$ có thể được trình bày là một quá trình Gausơ *phức* theo *t*. Tại bất kỳ thời điểm *t*, hàm mật độ xác suất của phần thực và phần ảo đều là Gausơ. Mô hình này cho thấy rằng với

mỗi τ hoặc τ_k, tia được tạo nên từ một số lượng lớn các thành phần không thể phân giải được. Vì vậy, $\tilde{c}(\tau,t)$ và $\tilde{a}_k(t)$ đều là quá trình Gausơ phức theo *t*.

Nếu $\tilde{c}(\tau,t)$ có trung bình không, thì đường bao $R = |\tilde{c}(\tau,t)|$ có hàm mật độ xác suất Rayleigh dạng:

$$f_R(r) = \frac{r}{\sigma^2} e^{-\frac{r^2}{2\sigma^2}}, \qquad r > 0 \tag{14.32}$$

Trong đó σ^2 là phương sai của các phần thực và phần ảo của $\tilde{c}(\tau,t)$.

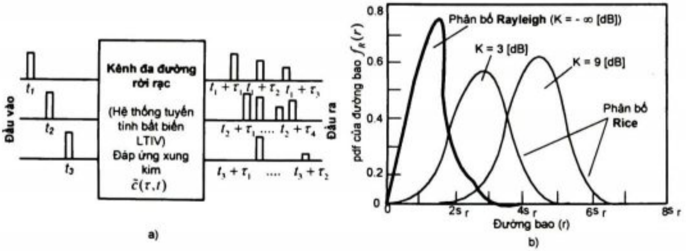
Nếu $\tilde{c}(\tau,t)$ có trung bình khác không (thể hiện sự tồn tại một đường truyền đi thẳng LOS), thì $R = |\tilde{c}(\tau,t)|$ có hàm mật độ xác suất phân bố Rice dạng:

$$f_R(r) = \frac{r}{\sigma^2} . I_0\left(\frac{Ar}{\sigma^2}\right) . e^{-\frac{r^2 + A^2}{2\sigma^2}}, \quad r > 0 \tag{14.33}$$

Trong đó *A* là trung bình khác không của $\tilde{c}(\tau,t)$ và $I_0(z)$ là hàm Besel cải tiến được định nghĩa bởi:

$$I_0(z) = \frac{1}{2\pi} \int_0^{2\pi} e^{z.\cos(u)} du \tag{14.34}$$

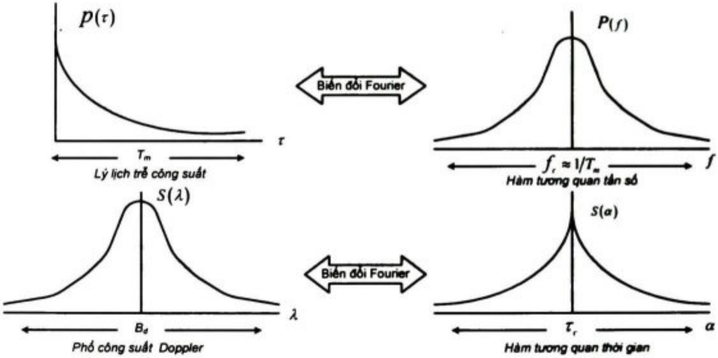
Tỉ lệ $K = \frac{A^2}{\sigma^2}$ được gọi là hệ số Rice thể hiện cho công suất tương đối trong các thành phần pha đinh và không pha đinh. Khi $K >> 1$ biểu thị pha đinh ít nghiêm trọng và khi $K << 1$ biểu thị pha đinh nghiêm trọng, khi $K = 0$ trở về kênh pha đinh Rayleigh. Hình 14.5b minh họa phân bố Rayleigh và Rice



Hình 14.5: a) Minh họa kênh pha đinh đa đường rời rạc *b) Minh họa phân bố Rayleigh và Rice*

Kênh được gọi là kênh pha đinh Rayleigh hay kênh pha đinh Rice phụ thuộc vào hàm mật độ xác suất pdf của $R = |\tilde{c}(\tau,t)|$. Các phân bố khác cho $|\tilde{c}(\tau,t)|$ như: Nakagami và Weibul cũng có thể có xẩy ra. Hàm mật độ xác suất tổng quát mô tả thống kê đường bao cho số lượng hữu hạn các thành phần phản chiếu, cùng với các đa đường khuếch tán gần đây được phát triển.

Trong các kết quả này, thì pha đinh Rayleigh và Rice là các trường hợp đặc biệt. Đối với kênh đa đường rời rạc, thì hàm mật độ xác suất pdf này ứng dụng cho $|\tilde{a}_k(t)|$. Trong khi pdf của $|\tilde{c}(\tau,t)|$ mô tả giá trị tức thời của đáp ứng xung kim phức, thì tính thay đổi theo thời gian được mô hình hóa bởi hoặc hàm tự tương quan hoặc hàm mật độ phổ công suất của quá trình ngẫu nhiên đó theo thời gian t. Dưới đây ta mô tả các mô hình này.



Hình 14.6: Quan hệ giữa các phần khác nhau của hàm tán xạ

14.6.1. Mô hình hóa tính thay đổi theo thời gian trong đáp ứng kênh

Bản chất thay đổi theo thời gian của các kênh được mô hình hóa bằng cách coi $\tilde{c}(\tau,t)$ là một quá trình ngẫu nhiên dừng nghĩa rộng WSS theo t có hàm tự tương quan:

$$R_{\tilde{c}\tilde{c}}(\tau_1,\tau_2,\alpha)=E\left\{\tilde{c}^*(\tau_1,t).\tilde{c}(\tau_2,t+\alpha)\right\} \tag{14.35}$$

Hầu hết các kênh đa đường, dịch pha và suy hao tương ứng với các trễ (nghĩa là các đường truyền) được giả định là không tương quan. Giả định tán xạ không tương quan US này dẫn đến:

$$R_{\tilde{c}\tilde{c}}(\tau_1,\tau_2,\alpha)=R_{\tilde{c}\tilde{c}}(\tau_1,\alpha).\delta(\tau_1-\tau_2) \tag{14.36}$$

Trong phương trình (14.6) bao gồm các giả định dừng nghĩa rộng WSS và tán xạ không tương quan US và thường được coi là mô hình WSSUS cho pha đinh và được đề xuất bởi Bello. Hàm tự tương quan này được ký hiệu là $R_{\tilde{c}\tilde{c}}(\tau,\alpha)$ và được cho bởi:

$$R_{\tilde{c}\tilde{c}}(\tau,\alpha)=E\left\{\tilde{c}^*(\tau,t).\tilde{c}(\tau,t+\alpha)\right\} \tag{14.37}$$

Bằng cách biến đổi Fourier hàm tự tương quan ta được mô hình trong miền tần số cho pha đinh dưới dạng mật độ phổ công suất là:

$$S(\tau,\lambda)=F\left\{R_{\tilde{c}\tilde{c}}(\tau,\alpha)\right\}=\int_{-\infty}^{\infty}R_{\tilde{c}\tilde{c}}(\tau,\alpha)e^{-j2\pi\lambda\alpha}d\alpha \tag{14.38}$$

Đại lượng $S(\tau,\lambda)$ được gọi là ***hàm tán xạ*** của kênh, là hàm của hai biến (biến miền thời gian-trễ và biến miền tần số-biến tần số doppler). Hàm tán xạ cho ta phép đo công suất đầu ra trung bình của kênh là một hàm của trễ và tần số doppler

Từ hàm tán xạ, ta có thể tìm được các thông số quan trọng của kênh, ảnh hưởng lên hiệu năng hệ của thống truyền thông làm việc trên kênh đó. Ta bắt đầu bằng lý lịch "*cường độ đa đường*" được định nghĩa là:

$$p(\tau) = R_{\tilde{c}\tilde{c}}(\tau,0) = E\left\{\left|\tilde{c}(\tau,t)\right|^2\right\} \tag{14.39}$$

Thể hiện cho công suất thu trung bình là một hàm của trễ τ. Phương trình (14.39) thường được gọi là lý lịch trễ công suất (*power-delay profile*). Thấy rõ, $p(\tau)$ quan hệ với hàm tán xạ qua:

$$p(\tau) = \int_{-\infty}^{\infty} S(\tau,\lambda)\,d\lambda \tag{14.40}$$

Một hàm hữu hiệu khác để đặc trưng hóa pha đinh là phổ công suất Doppler được rút ra từ hàm tán xạ theo:

$$S_d(\lambda) = \int_{-\infty}^{\infty} S(\tau,\lambda)\,d\tau \tag{14.41}$$

Các quan hệ giữa các hàm này được cho ở hình 14.6.

Lý lịch cường độ đa đường thường được đo bằng cách dò kênh theo dạng sóng RF băng rộng trong đó tín hiệu điều chế là chuỗi PN tốc độ cao. Có thể tìm công suất theo lý lịch trễ bằng cách lấy tương quan đầu ra máy thu với các phiên bản bị trễ của chuỗi PN, và đo giá trị trung bình đầu ra bộ lấy tương quan. Nơi, các phép đo cho các ứng dụng vô tuyến di động với một trạm BS cố định và người dùng di động được xét, thì lý lịch trễ công suất được đo trong các gia số khoảng cách ngắn của phân số bước sóng. Sau đó, lý lịch công suất đó được trung bình hóa trên (10 đến 20) bước sóng để lấy trung bình các ảnh hưởng của kênh pha đinh Rayleigh. Các phép đo tương quan được thực hiện như một hàm của vị trí, nghĩa là hàm tự tương quan không gian có thể được chuyển thành hàm tương quan thời gian bằng cách lưu ý rằng $\Delta X = v.\Delta t$, trong đó ΔX là sự di chuyển không gian gia tăng của di động và v là tốc độ. Vì vậy, có thể tìm được phổ Doppler bằng cách chuyển hàm tương quan thời gian đối với bất kỳ tốc độ xe cộ.

14.6.2. Các tham số quan trọng

Hàm tán xạ, lý lịch cường độ đa đường và phổ doppler mô tả chi tiết các khía cạnh khác nhau của kênh pha đinh. Tuy nhiên, hai tham số quan trọng nhất để mô phỏng kênh pha đinh là trải rộng đa đường và độ rộng băng tần Doppler.

Trải rộng đa đường

Các bộ biểu thị quan trọng về tính nghiêm trọng của các ảnh hưởng đa đường là trải trễ cực đại và trải trễ trung bình quan phương **rms**. Trải trễ cực đại là giá trị trễ T_{max} tại đó công suất thu $p(\tau)$ là rất nhỏ và trải trễ trễ rms σ_τ được định nghĩa là:

$$\sigma_\tau = \sqrt{<\tau^2> - <\tau>^2} \tag{14.42}$$

Trong đó < x > ký hiệu cho giá trị trung bình theo thời gian của x và:

$$< \tau^k > = \frac{\int \tau^k p(\tau) d\tau}{\int p(\tau) d\tau} \quad (14.43)$$

Kênh chọn lọc tần số

Khi trải trễ là bậc (nghĩa là lớn hơn) khoảng thời gian của một ký hiệu thì các thành phần đa đường bị trễ sẽ đến trong các khoảng thời gian ký hiệu khác và gây ra giao thoa giữa các ký hiệu ISI làm ảnh hưởng hiệu năng BER hệ thống. Điều này tương đương với hàm truyền đạt thay đổi theo thời gian của kênh có độ rộng băng thông nhỏ hơn độ rộng băng tần của tín hiệu. Khi này, kênh thực hiện như một bộ lọc giới hạn băng tần và được gọi là kênh chọn lọc tần số.

$$T_{max} >> T_s \quad hoÆc \quad \sigma_T > 0,1T_s \quad (14.44)$$

Quan hệ vào/ra là tích chập:

$$\tilde{y}(t) = \tilde{c}(\tau, t) \otimes \tilde{x}(t) \quad (14.45)$$

Trong đó ⊗ ký hiệu phép tích chập. Trong khi trải trễ (cực đại hoặc rms) ảnh hưởng lớn lên hiệu năng hệ thống, thì có thể thấy rằng hiệu năng hệ thống không quá nhạy cảm với dạng của lý lịch cường độ đa đường. Dạng được giả định phổ biến nhất của $p(\tau)$ là đều và số mũ.

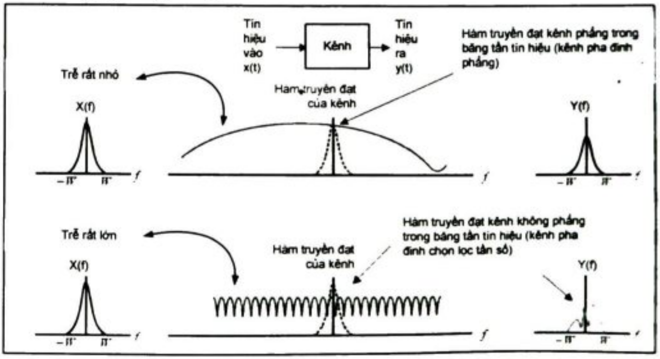
Kênh không chọn lọc tần số

Đối với kênh không chọn lọc tần số, thì trải trễ cự đại T_{max} nhỏ hơn nhiều so với khoảng thời gian kí hiệu T_S (nghĩa là $T_{max} << T_S$).

$$T_{max} << T_S \text{ hoặc } \sigma_T < 0,1.T_S \quad (14.46)$$

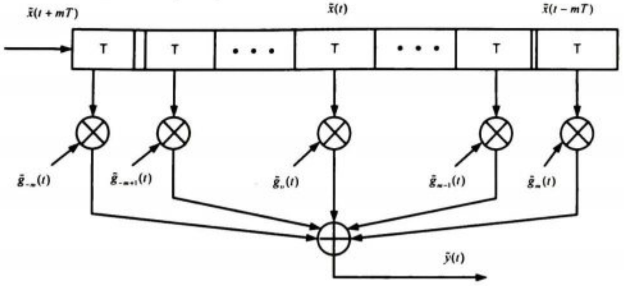
Khi này, tất cả các thành phần đa đường bị trễ đến máy thu trong phần nhỏ thời gian một ký hiệu và kênh có thể được mô hình hóa bằng một tia, và quan hệ vào/ra được biểu diễn là một phép nhân. Nói cách khác:

$$\tilde{y}(t) = \tilde{a}(t) \times \tilde{x}(t) \quad (14.47)$$



Hình 14.7(a): Minh họa khái niệm kênh pha đinh phẳng và chọn lọc tần số

Hình 14.7(a) Minh họa khái niệm kênh pha đinh chọn lọc tần số và kênh pha đinh không chọn lọc tần số (kênh pha đinh phẳng).



Hình 14.7(b): Mô hình TDL cho kênh đa đường khuếch tán với $g_n(t) = T.\tilde{c}(nT,t)$

Độ rộng băng tần Doppler

Độ rộng băng tần Doppler (hoặc trải rộng doppler) B_d là độ rộng băng tần của phổ Doppler được định nghĩa bởi (14.41), và là bộ biểu thị mức độ thay đổi các đặc tính kênh (pha đinh) là một hàm của thời gian. Nếu B_d là bậc của độ rộng băng tần của tín hiệu, thì các đặc tính kênh thay đổi (pha đinh) tại một tốc độ có thể so sánh được với tốc độ ký hiệu và kênh được gọi là pha đinh nhanh. Ngược lại được gọi là pha đinh chậm. Vì vậy:

$$\begin{aligned} &B_d << B_S \approx \frac{1}{T_S}, \quad \text{Kênh pha đinh chậm} \\ &B_d >> B_S \approx \frac{1}{T_S}, \quad \text{Kênh pha đinh nhanh} \end{aligned} \tag{14.48}$$

Nếu kênh là kênh pha đinh chậm, thì có thể dùng giải pháp chớp ảnh để mô phỏng kênh khi ước tính hiệu năng. Ngược lại, các thay đổi động trong các điều kiện kênh phải được mô phỏng một cách rõ ràng.

14.7. Phương pháp luận mô phỏng

Ta xét mô phỏng các kênh pha đinh đa đường. Ta giả thiết rằng, mô hình được xét hoặc là mô hình kênh đa đường khuếch tán hoặc là mô hình kênh đa đường rời rạc và chúng đều là tán xạ không tương quan dừng nghĩa rộng WSSUS. Cũng giả thiết rằng, biết trước các phân bố, lý lịch trễ và phổ Doppler. Hơn nữa, ta cho rằng pha đinh là Rayleigh hoặc Ricean, nhấn mạnh vào pha đinh Rayleigh vì có thể tìm được mô hình Ricean từ mô hình Rayleigh bằng cách xét trung bình khác không. Ta bắt đầu bằng kênh đa đường khuếch tán.

14.7.1. Mô phỏng kênh pha đinh đa đường khuếch tán

Kênh đa đinh đa đường khuếch tán là một hệ thống tuyến tính thay đổi theo thời gian LTV được đặc trưng hóa bởi đáp ứng xung kim thay đổi theo thời gian t $\tilde{c}(\tau,t)$ *liên tục* chứ không phải là rời rạc. Mô hình mô phỏng cho hệ thống LTV được rút ra ở chương 13 và ở đây

ta chỉ nhắc lại các bước cần thiết. Vì giả định đầu vào thông thấp của kênh được hạn chế băng thông là B ($B \approx r$ cho trường hợp băng thông, r là tốc độ ký hiệu), nên ta có thể biểu diễn đầu vào thông thấp theo các giá trị mẫu của nó dùng tốc độ lấy mẫu nhỏ nhất là r mẫu trên giây như:

$$\tilde{x}(t-\tau)=\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\frac{\sin\left(2\pi B(\tau-nT)\right)}{2\pi B(\tau-nT)} \tag{14.49}$$

Trong đó $T = 1/r$ là thời gian giữa hai mẫu.

Thế $\tilde{x}(t-\tau)$ vào tích chập.

$$\tilde{y}(t)=\int_{+\infty}^{\infty}\tilde{c}(\tau,t)\tilde{x}(t-\tau)d\tau \tag{14.50}$$

Ta được:

$$\begin{aligned}\tilde{y}(\tau)&=\int_{-\infty}^{\infty}\tilde{c}(\tau,t)\left\{\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\frac{\sin\left(2\pi B(\tau-nT)\right)}{2\pi B(\tau-nT)}\right\}d\tau\\&=\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\int_{-\infty}^{\infty}\tilde{c}(\tau,t)\left\{\frac{\sin(2\pi B(\tau-nT))}{2\pi B(\tau-nT)}\right\}d\tau\end{aligned} \tag{14.51}$$

Vì vậy:

$$\tilde{y}(\tau)=\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\tilde{g}_n(t) \tag{14.52}$$

Trong đó:

$$\tilde{g}_n(t)=\int_{-\infty}^{\infty}\tilde{c}(\tau,t)\left\{\frac{\sin(2\pi B(\tau-nT))}{2\pi B(\tau-nT)}\right\}d\tau \tag{14.53}$$

Các mô hình mô phỏng cho các kênh pha đinh đa đường khuếch tán được rút ra từ (14.52) sử dụng hai phép lấy xấp xỉ. Cắt xén tổng trong (14.52) sao cho chỉ còn các thành phần trong khoảng $/n/ < m$ và lấy xấp xỉ tích phân trong (14.53) là:

$$\tilde{g}_n(t)\approx T.\tilde{c}(nT,t) \tag{14.54}$$

Dẫn đến dạng tính toán hiệu quả:

$$\tilde{y}(t)=\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\tilde{g}_n(t)\approx\sum_{n=-m}^{m}\tilde{x}(t-nT)\tilde{g}_n(t)\approx T\sum_{n=-m}^{m}\tilde{x}(t-nT)\tilde{c}(nT,t) \tag{14.55}$$

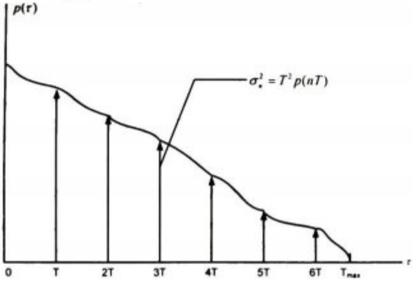
Thực thi hóa phương trình (14.55) bằng cách sử dụng đường trễ rẽ nhánh được ở hình 14.7(b).

$$\tilde{y}(t)=\sum_{n=-\infty}^{\infty}\tilde{x}(t-nT)\tilde{g}_n(t)\approx\sum_{n=-m}^{m}\tilde{x}(t-nT)\tilde{g}_n\approx T\sum_{n=-m}^{m}\tilde{x}(t-nT)\tilde{c}(nT,t)$$

Đối với kênh pha đinh Rayleigh, các quá trình độ lợi nhánh $g_n(t)\approx T.\tilde{c}(nT,t)$ là các quá trình Gausơ phức trung bình không. Chúng sẽ không tương quan do giả định WSSUS. Mật độ phổ công suất của mỗi quá trình độ lợi nhánh được định rõ bởi phổ Doppler và phương sai σ_n^2 của quá trình độ lợi nhánh thứ n được cho bởi:

$$E\left\{\left|\tilde{g}_n(t)\right|^2\right\}\approx\sigma_n^2=T^2E\left\{\left|\tilde{c}(nT,t)\right|^2\right\}=T^2p(nT) \tag{14.56}$$

Và tìm được từ các giá trị mẫu của lý lịch cường độ đa đường $p(\tau)$, một ví dụ được cho ở hình 14.8 có tổng số nhánh là T_{max}/T.



Hình 14.8: Các giá trị mẫu của lý lịch trễ công suất

Các trường hợp đặc biệt

Nếu là kênh bất biến theo thời gian, thì $\tilde{c}(\tau,t) = \tilde{c}(\tau)$ và các độ lợi nhánh trở thành hằng số, vì vậy:

$$\tilde{g}_x(t) = \tilde{g}_n \approx T.\tilde{c}(nT) \tag{14.57}$$

Nói cách khác, các độ lợi nhánh là các giá trị mẫu của đáp ứng xung kim của hệ thống tuyến tính bất biến LTIV, và mô hình đường trễ rẽ nhánh rút gọn thành bộ lọc FIR thực hiện tích chập trong trong miền thời gian.

Nếu là kênh không chọn lọc tần số, thì chỉ có một nhánh trong mô hình đó và $\tilde{y}(t) = \tilde{x}(t).\tilde{g}(t)$, trong đó $\tilde{g}(t)$ hoặc là một quá trình Rayleigh hoặc là một quá trình Ricean.

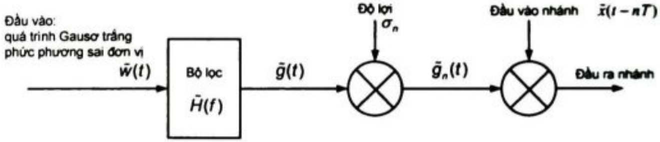
Lấy mẫu

Một khía cạnh quan trọng khác của mô hình đường trễ rẽ nhánh TDL đáng lưu ý là tỉ lệ lấy mẫu để mô phỏng. Mô hình TDL được cho ở hình 14.8 được rút ra với đầu vào/ra $\widetilde{x}(t)/\widetilde{y}(t)$ thời gian liên tục. Tuy nhiên, trong mô phỏng ta sử dụng các giá trị mẫu của các đầu vào/ra tương ứng $\widetilde{x}(t)/\widetilde{y}(t)$, thường lấy mẫu tại tốc độ (8 đến 32) lần độ rộng băng tần, trong đó độ rộng băng tần phải tính đến ảnh hưởng trải rộng do bản chất thay đổi theo thời gian của hệ thống như được đề cập ở chương 13. Lưu ý rằng, tốc độ Nyquist là $2B$, $B = r/2$ đã được sử dụng để rút ra mô hình TDL, và khoảng cách nhánh là $T=1/r$ sẽ lớn hơn rất nhiều so với thời gian lấy mẫu dạng sóng vào/ra T_S. Tất nhiên, ta có thể rút ra mô hình TDL có khoảng cách nhánh nhỏ hơn (nghĩa là nhiều mẫu/ký hiệu hơn) nhưng không hiệu quả tính toán và không cần thiết cải thiện độ chính xác của mô phỏng.

Tạo các quá trình độ lợi nhánh

Các quá trình độ lợi nhánh là các quá trình ngẫu dừng có các hàm mật độ xác suất Gausơ và các hàm mật độ phổ công suất tùy ý. Mô hình đơn giản nhất cho các quá trình độ lợi nhánh

là coi chúng là các quá trình Gausơ trung bình không, phức, không tương quan có các phương sai khác nhau nhưng các mật độ phổ công suất đồng nhất. Khi này, các quá trình độ lợi nhánh có thể được tạo ra bằng lọc các quá trình Gausơ trắng như cho ở hình 14.9.



Hình 14.9: Tạo quá trình độ lợi nhánh thứ n

Hàm truyền đạt bộ lọc được chọn sao cho nó tạo ra mật độ phổ công suất doppler mong muốn. Nói cách khác, H(f) được chọn sao cho:

$$S_{\tilde{g}\tilde{g}}(f) = S_d(f) = S_{\tilde{w}\tilde{w}}(f).\left|\tilde{H}(f)\right|^2 = \left|\tilde{H}(f)\right|^2 \tag{14.58}$$

Trong đó $S_{\tilde{w}\tilde{w}}(f)$ là mật độ phổ công suất của quá trình tạp âm trắng đầu vào, được đặt bằng 1 và $S_{\tilde{g}\tilde{g}}(f)$ là mật độ phổ công suất cụ thể của các quá trình độ lợi nhánh. Độ lợi của bộ lọc được chọn sao cho $\tilde{g}(t)$ có công suất chuẩn hóa là 1. Độ lợi tĩnh σ_n trong hình 14.9 giải thích các mức công suất hay phương sai khác nhau cho các nhánh khác nhau. Nếu mật độ phổ công suất của các độ lợi nhánh là khác nhau, thì ta sẽ dùng các bộ lọc khác nhau cho các nhánh khác nhau.

Các lý lịch trễ công suất và các mật độ phổ công suất Doppler

Như đã đề cập ở trên, hiệu năng BER của hệ thống nhạy cảm với các giá trị trải trễ cực đại và rms hơn so với dạng của lý lịch trễ công suất. Theo đó, ta dùng các lý lịch đơn giản như dạng đồng đều hoặc mũ để mô phỏng. Các lý lịch trễ được chuẩn hóa để có diện tích đơn vị (nghĩa là, công suất tổng được chuẩn hóa, hoặc diện tích dưới lý lịch trễ công suất trung bình cục bộ được đặt bằng 1). Vì vậy:

$$\int_0^{T_m} p(\tau)dt = 1 \tag{14.59}$$

Các trải trễ **rms** điển hình được cho ở bảng 14.2.

Bảng 14.2: Các trải trễ rms điển hình

Kiểu tuyến	Khoảng cách tuyến	Trải trễ rms
Tán xạ đối lưu	100 km	ms (10^{-3})
Di động ngoài trời	1 km	μs (10^{-6})
Trong nhà	10 m	ns (10^{-9})

Các mô hình được sử dụng phổ biến nhất cho các mật độ phổ công suất Doppler đối với các ứng dụng di động coi rằng có nhiều thành phần đa đường, mỗi thành phần có các trễ khác nhau và tất cả các thành phần đều có cùng phổ Doppler. Thực tế, mỗi thành phần đa đường (tia)

tạo ra một số lượng lớn thành phần đa đường không thể phân giải được cùng lúc, có góc tới phân bố đều tại ăng ten thu. Mô hình kênh này được sử dụng bởi Jakes và một số khác ở phòng thí nghiệm Bell để rút ra mô hình kênh vô tuyến di động toàn diện đầu tiên cho các hiệu ứng Doppler và các hiệu ứng pha đinh biên độ. Phổ Doppler Jake kinh điển có dạng, đã được mô phỏng khởi đầu trong chương 7 (xem ví dụ 7.11):

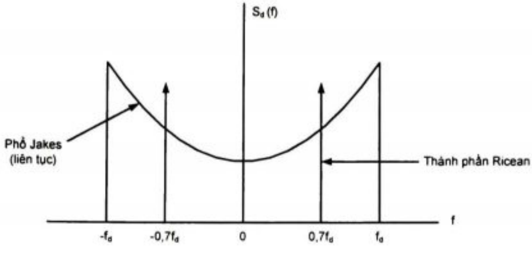
$$S_d(f) = S_{\tilde{g}_n\tilde{g}_n}(f) = \frac{K}{\sqrt{1-(f/f_d)^2}}, \qquad -f_d \le f \le f_d \tag{14.60}$$

Trong đó $f_d = v/\lambda$ là dịch doppler cực đại, v là vận tốc xe cộ m/s, λ là bước sóng của sóng mang. Trong khi phổ Doppler được định nghĩa bởi (14.60) là phù hợp cho môi trường tán xạ dày đặc như các khu vực thành phố, thì "*phổ Ricean*" được khuyến nghị cho môi trường nông thôn ở đó có một thành phần đi thẳng trực tiếp LOS mạnh vì vậy pha đinh Ricean. Phổ Doppler Ricean có dạng:

$$S_d(f) = S_{\tilde{g}_n\tilde{g}_n}(f) = \frac{0,41}{\sqrt{1-(f/f_d)^2}} + 0,91\delta(f \pm 0,7f_d), \quad -f_d \le f \le f_d \tag{14.61}$$

Được thấy ở hình 14.10. Các dạng phổ khác được dùng cho mật độ phổ công suất Doppler bao gồm Gausơ và đều. Các độ rộng băng tần Doppler điển hình trong các ứng dụng di động ở 1GHz sẽ nằm trong khoảng từ 10 đến 200 Hz.

Có nhiều cách thực hiện lọc định dạng phổ Doppler để tạo các quá trình độ lợi nhánh trong mô hình TDL cho kênh khi dùng mô hình Jakes. Bộ lọc FIR trong miền thời gian là phổ biến nhất vì mật độ phổ công suất Doppler không dễ dàng thực thi ở dạng đệ quy. Tạo phổ Jakes dùng các kỹ thuật lọc FIR đã được minh họa trong chương 7.



Hình 14.10: Minh họa các mật độ phổ công suất doppler

Cần lưu ý rằng, khi tạo các quá trình độ lợi nhánh, độ rộng băng tần của các quá trình độ lợi nhánh cho các kênh thay đổi chậm theo thời gian sẽ rất nhỏ so với độ rộng băng tần của tín hiệu đi qua chúng. Khi này, ta nên thiết kế bộ lọc độ lợi nhánh và được thực hiện tại tốc độ lấy mẫu thấp hơn. Có thể dùng nội suy tại đầu ra của bộ lọc để tạo các mẫu dày đặc hơn ở tốc độ phù hợp với tốc độ lấy mẫu tín hiệu đến nhánh đó. Thiết kế bộ lọc tại tốc độ cao hơn sẽ dẫn đến không hiệu quả tính toán và các vấn đề ổn định.

Mô hình độ lợi nhánh tương quan

Việc lấy xấp xỉ các quá trình độ lợi nhánh được cho ở các biểu thức (14.53) và (14.54) bởi:

$$\tilde{g}_n(t) = \int_{-\infty}^{\infty} \tilde{c}(\tau,t)\left\{\frac{\sin\left(2\pi B(\tau-nT)\right)}{2\pi B\left(\tau-nT\right)}\right\}d\tau \approx T.\tilde{\mathcal{C}}\left(nT,t\right) \qquad (14.62)$$

Dẫn đến các hàm độ lợi nhánh không tương quan. Không lấy xấp xỉ, thì các hàm độ lợi nhánh sẽ *tương quan*. Thấy rõ, tương quan giữa $g_n(t)$ và $g_m(t)$ được cho bởi:

$$\begin{aligned} R_{m,n}(\eta) &= E\left\{g_m^*(t)g_n(t+\eta)\right\} \\ &= \int R_{\tilde{c}\tilde{c}}(\tau,\eta).\sin C\left(2B\tau-m\right).\sin C\left(2B\tau-n\right)d\tau \end{aligned} \qquad (14.63)$$

Trong đó $T = 1/2B$ là *khoảng cách nhánh*.

Việc tạo một tập các quá trình ngẫu nhiên tương quan có các hàm mật độ phổ công suất tùy ý là rất khó khăn. Việc lấy xấp xỉ làm đơn giản bài toán một chút, tạo ra giả định hợp lý để tất cả các hàm độ lợi nhánh có cùng mật độ phổ công suất. Vì vậy, ta giả thiết rằng:

$$S(\tau,\lambda) = m(\tau)S_d(\lambda) \qquad (14.64)$$

Trong đó $S(\tau,\lambda)$ là hàm tán xạ, $m(\tau)$ là lý lịch công suất trễ chuẩn hóa và $S_d(\lambda)$ là phổ Doppler.

14.7.2. Mô phỏng kênh pha đinh đa đường rời rạc

So sánh với mô hình đa đường *khuếch tán* thì việc mô phỏng các mô hình đa đường rời rạc là dễ hiểu hơn, ít nhất là khái niệm. Ta phải luôn nhớ vì tính cách động của kênh theo cả thời gian và không gian, nên cần phải cẩn thận để tránh chồng phổ. Quan hệ vào/ra của mô hình kênh rời rạc được cho bởi:

$$\tilde{y}(t) = \sum_{k=1}^{N(t)} \tilde{a}_k(t).\tilde{x}\left(t-\tau_k(t)\right) \qquad (14.65)$$

Trong đó $\tilde{a}_k(t)$ là suy hao đường dẫn phức như đã thảo luận trong phần 14.4. Trong (14.65), coi rằng số lượng các thành phần đa đường và cấu trúc trễ sẽ thay đổi chậm so với các thay đổi trong $\tilde{a}_k(t)$. Vì vậy, trễ $\tau_k(t)$ có thể coi là không đổi trong khoảng thời gian mô phỏng và phương trình trên được viết như sau:

$$\tilde{y}(t) = \sum_{k=1}^{N(t)} \tilde{a}_k(t).\tilde{x}\left(t-\tau_k^{*}\right) \qquad (14.66)$$

Và được thực thi dưới dạng sơ đồ khối như được cho ở hình 14.11.

Để minh họa giải pháp cơ bản nhằm mô phỏng các mô hình kênh rời rạc, ta coi mô hình được định rõ dưới dạng các phân bố xác suất cho *N* thành phần trễ, và suy hao phức như một hàm của trễ. Trình bày (chớp ảnh) của kênh như sau:

1. Lấy ra một con số ngẫu nhiên *N* để có được số lượng trễ.

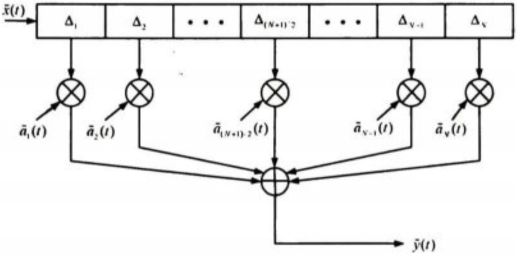
2. Rút ra một tập *N* số ngẫu nhiên từ phân bố cho các giá trị trễ.

3. Lấy ra một tập N suy hao dựa trên các giá trị trễ.

Tập $3N$ số ngẫu nhiên này thể hiện một chớp ảnh của kênh, được thực thi như được cho ở hình 14.11. Trong hình 14.11, trễ khởi đầu $\Delta_1 = \tau_1$. Các trễ còn lại Δ_n, $2 \le n \le N$ là các trễ vi sai, được định nghĩa bởi:

$$\Delta_n = \tau_n - \tau_{n-1}, \qquad 2 \le n \le N \tag{14.67}$$

Trong khi việc thực thi được cho ở hình 14.11 là khá dễ hiểu, thì nó nảy sinh vấn đề khi các trễ chỉ lệch thời gian rất nhỏ. Vì tất cả sẽ được lấy mẫu, các khoảng cách nhánh (nghĩa là, hiệu số trễ $\tau_n - \tau_{n-1}$) phải được biểu diễn theo số nguyên lần chu kỳ lấy mẫu trong mô phỏng. Vì vậy, thời gian lấy mẫu phải rất nhỏ, nhỏ hơn hiệu số trễ nhỏ nhất, dẫn đến tốc độ lấy mẫu quá mức và gánh nặng tính toán là không chấp nhận được. Ta có thể tránh vấn đề này bằng cách triển khai mô hình TDL với khoảng cách nhánh cách đều theo giải pháp được sử dụng trong mô phỏng cho các kênh đa đường khuếch tán ở phần 14.7.1.



Hình 14.11: Mô hình TDL trễ khả biến cho các kênh đa đường rời rạc

Mô hình TDL cách đều cho các kênh pha đinh đa đường rời rạc

Các độ lợi nhánh của mô hình TDL cách đều được cho trong (14.53) là:

$$\tilde{g}_n(t) = \int_{-\infty}^{\infty} \tilde{c}(\tau,t)\left\{\frac{\sin\left(2\pi B(\tau - nT)\right)}{2\pi B(\tau - nT)}\right\} d\tau \tag{14.68}$$

Thế đáp ứng xung kim của kênh đa đường rời rạc, được cho bởi:

$$\tilde{c}(\tau,t) = \sum_{k=1}^{N(t)} \tilde{a}_k(t).\delta(t - \tau_k) \tag{14.69}$$

Vào phương trình 14.68, ta được các độ lợi nhánh là:

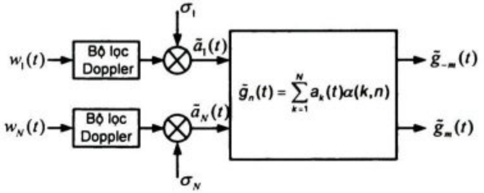
$$\tilde{g}_n(t) = \sum_{k=1}^{N} \tilde{a}_k(t).\sin c\left(\frac{\tau_k}{T} - n\right) = \sum_{k=1}^{N} \tilde{a}_k(t).\alpha(k,n) \tag{14.70}$$

Trong (14.70):

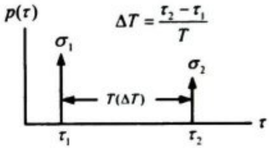
$$\alpha(k,n) = \sin c\left(\frac{\tau_k}{T} - n\right) \tag{14.71}$$

Lưu ý rằng đường bao của $\alpha(k,n)$ giảm khi $|n|$ tăng. Vì vậy, số lượng nhánh có thể được cắt gọn trong khoảng $|n| \le m$, trong đó m được chọn để thỏa mãn $m >> T_{max}T$. Với trường hợp, trải trễ cực đại T_{max} không vượt quá 3 hoặc 4 lần ký hiệu, thì số lượng nhánh cần thiết không lớn hơn 20 ($-m < n < m$, $m = 10$). Tại đây, mô hình có dạng như đã được rút ra cho mô hình đa đường khuếch tán được minh họa ở hình 14.7.

Quá trình tạo các độ lợi nhánh được minh họa ở hình 14.12. Lưu ý rằng, việc tạo các quá trình độ lợi nhánh cho mô hình đa đường rời rạc là dễ hiểu so với việc tạo quá trình độ lợi nhánh cho trường hợp khuếch tán. Ta bắt đầu với tập N quá trình tạp âm trắng Gausơ phức trung bình không độc lập thống kê, chúng được lọc để tạo phổ Doppler phù hợp. Sau đó, được lấy tỉ lệ để tạo lý lịch công suất mong muốn và cuối cùng được chuyển đổi theo (14.70) để tạo các quá trình độ lợi nhánh (Lưu ý rằng chỉ có hai trong số N đường truyền được thấy trong hình 14.12).



Hình 14.12: Tạo các quá trình độ lợi nhánh



Hình 14.13: Mô hình hai tia đơn giản $\Delta\tau$ được chuẩn hóa

Để minh họa quá trình tính toán các hàm độ lợi nhánh, ta giả thiết:

$$\Delta\tau = \frac{\tau_2 - \tau_1}{T} = 0,5 \tag{14.72}$$

Trong trường hợp này, tìm được các hàm độ lợi nhánh bằng cách lọc hai quá trình tạp âm Gausơ trắng không tương quan sau đó biến đổi chúng thành các quá trình độ lợi nhánh theo (14.70) như sau:

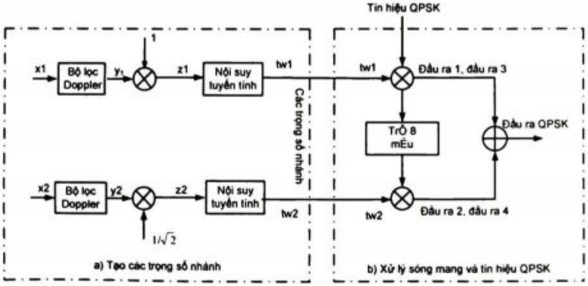
$$\begin{bmatrix} \tilde{g}_{-4}(t) \\ \tilde{g}_{-3}(t) \\ \tilde{g}_{-2}(t) \\ \tilde{g}_{-1}(t) \\ \tilde{g}_{0}(t) \\ \tilde{g}_{1}(t) \\ \tilde{g}_{2}(t) \\ \tilde{g}_{3}(t) \\ \tilde{g}_{4}(t) \end{bmatrix} = \begin{bmatrix} \sin c(0,0+4) & \sin c(0,5+3) \\ \sin c(0,0+3) & \sin c(0,5+3) \\ \sin c(0,0+2) & \sin c(0,5+2) \\ \sin c(0,0+1) & \sin c(0,5+1) \\ \sin c(0,0) & \sin c(0,5) \\ \sin c(0,0-1) & \sin c(0,5-1) \\ \sin c(0,0-2) & \sin c(0,5-2) \\ \sin c(0,0-3) & \sin c(0,5-3) \\ \sin c(0,0-4) & \sin c(0,5-4) \end{bmatrix} \times \begin{bmatrix} \tilde{a}_1(t) \\ \tilde{a}_2(t) \end{bmatrix} \tag{14.73}$$

Nó là:

$$\begin{bmatrix} \tilde{g}_{-4}(t) \\ \tilde{g}_{-3}(t) \\ \tilde{g}_{-2}(t) \\ \tilde{g}_{-1}(t) \\ \tilde{g}_{0}(t) \\ \tilde{g}_{1}(t) \\ \tilde{g}_{2}(t) \\ \tilde{g}_{3}(t) \\ \tilde{g}_{4}(t) \end{bmatrix} = \begin{bmatrix} 0,0 & 0,0707 \\ 0,0 & -0,0910 \\ 0,0 & 0,1273 \\ 0,0 & -0,2122 \\ 1,0 & 0,6366 \\ 0,0 & 0,6366 \\ 0,0 & -0,2122 \\ 0,0 & 0,1273 \\ 0,0 & -0,0909 \end{bmatrix} \times \begin{bmatrix} \tilde{a}_1(t) \\ \tilde{a}_2(t) \end{bmatrix} \qquad (14.74)$$

Trong đó $\widetilde{a}_1$ và $\widetilde{a}_2$ được định nghĩa trong hình 14.12. Phương trình trên cho thấy các hệ số của phép biến đổi chỉ là 9 nhánh. Ta thấy rằng những hệ số này sẽ là không đáng kể với độ lợi nhánh bậc cao hơn và kết quả là chúng có thể được bỏ qua. Vì thế, mô hình TDL có thể rút gọn lại 9 nhánh.

Thường dùng mô hình hai tia để dự đoán hiệu năng sơ bộ cho các kênh pha đinh. Xét lý lịch trễ công suất được minh họa ở hình 14.13. Có thể dựa đoán hiệu năng tham số bằng cách thay đổi tỉ lệ của trải trễ chuẩn hóa $\Delta\tau = (\tau_2 - \tau_1)/T$, T là khoảng thời gian ký hiệu và tỉ lệ công suất tương đối trong hai đường $(\sigma_1/\sigma_2)^2$. Nếu $\Delta\tau << 0,1$ thì có thể kết hợp hai đường lại thành một và mô hình có thể được coi là *không chọn lọc tần số*. Nếu $\Delta\tau > 0,1$ thì sẽ có ISI đáng kể trong kênh và kênh được coi là *chọn lọc tần số*.



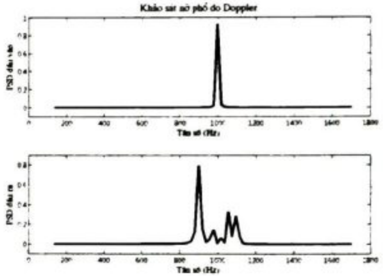
Hình 14.14: Các sơ đồ khối của các hệ thống mô phỏng

Ví dụ 14.2: Trong ví dụ này, ta xét ảnh hưởng pha đinh do Doppler lên truyền dẫn tín hiệu QPSK trên kênh đa đường rời rạc. Sơ đồ khối được cho ở hình 14.14. Việc tạo các trọng số nhánh được cho ở hình 14.14(a). Thực thi bộ lọc doppler bằng cách dùng mô hình Jakes được định nghĩa bởi biểu thức (14.60) với $K = 1$ và $f_d = 100$ Hz. Các quá trình độ lợi nhánh là không tương quan nhau và Gausơ. Khoảng cách nhánh dựa vào độ rộng băng tần 20 kHz (độ rộng băng tần tương đương thông thấp là10 kHz). Các trọng số nhánh được ký hiệu là *tw1* và *tw2*.

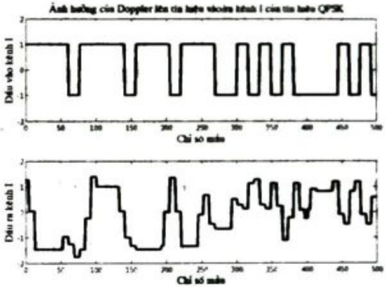
Tín hiệu phức được nhân với các trọng số nhánh phức. Cả tín hiệu QPSK phức và sóng mang phức được sử dụng làm các đầu vào. Sóng mang được định nghĩa bởi:

$$c(t) = e^{j2\pi(1000)t} \tag{14.75}$$

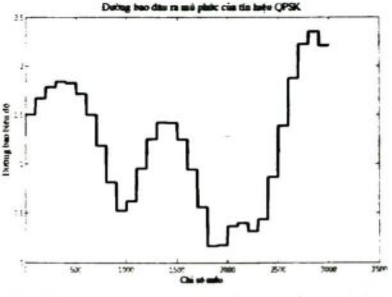
Trễ của 8 mẫu tương đương với một nửa của thời gian ký hiệu. Chi tiết thêm được cho ở trong mã chương trình Matlab **NVD14_jakes.m** ở phụ lục 14A.



Hình 14.15: Mật độ phổ công suất đầu vào (trên) và đầu ra (dưới)



Hình 14.16: Đầu vào và đầu ra QPSK kênh đồng pha

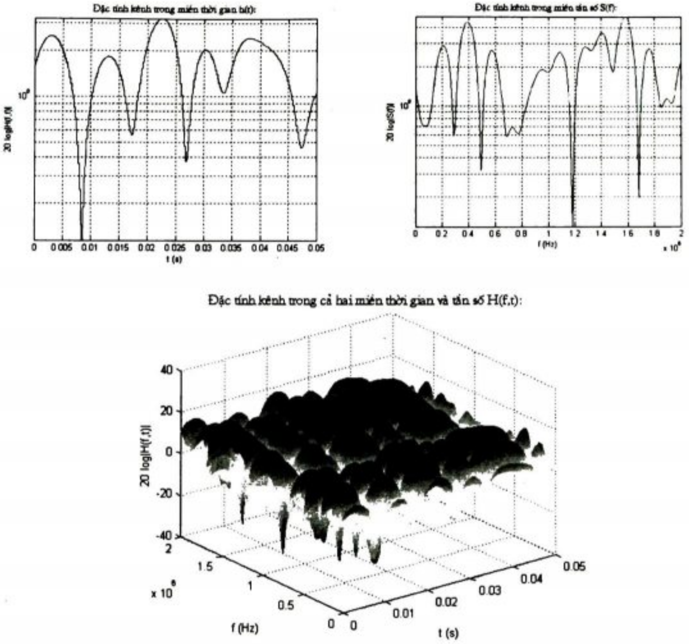


Hình 14.17(a): Đường bao của đầu ra số mũ phức

Độ dài mô phỏng được xác định từ số lần xét. Để quan trắc phổ đầu vào/ra cho trường hợp số mũ phức, cần có 10 đến 20 chu kỳ của số mũ phức. Tại cùng thời điểm, để bắt giữ được các ảnh hưởng của kênh thay đổi theo thời gian, ta cần phải mô phỏng quá trình pha đinh trong khoảng thời gian (5 đến 10) lần nghịch đảo của độ rộng băng tần Doppler. Hai vấn đề này dẫn đến thời gian mô phỏng là 1/20 giây hoặc 8,000 mẫu.

Chạy chương trình Matlab **NVD14_jakes.m** được cho ở phụ lục 14A nhận được các kết quả trong hình 14.15, 14.16 và 14.17. Các tín hiệu sóng mang đầu vào/ra được cho ở hình 14.15. Đầu vào (trên) là tín hiệu tại 1000 Hz, đầu ra (dưới) minh họa dãn phổ do Doppler. Đầu vào/ra kênh đồng pha được minh họa ở hình 14.16. Tín hiệu đầu vào (trên) có hai mức như

mong đợi. Tín hiệu đầu ra (dưới) có nhiều hơn hai mức vì giao thoa giữa các ký hiệu ISI. Đường bao của tín hiệu đầu ra QPSK được minh họa trên hình 14.17(a). Đặc tính kênh trong các miền thời gian và tần số được minh họa ở hình 14.17(b).



Hình 14.17(b): Minh họa đặc tính kênh trong các miền thời gian và tần số

14.7.3. Các mô hình kênh pha đinh đa đường rời rạc điển hình

Phần này, trình bày các ví dụ mô hình đa đường rời rạc để mô phỏng hiệu năng các hệ thống truyền thông vô tuyến.

Mô hình đầu tiên, được gọi là mô hình Rummler cho các tuyến vi ba mặt đất giữa các tháp ăng ten cố định. Đây là kênh vô tuyến tầm nhìn thẳng LOS có rất ít thành phần đa đường vì sử dụng ăng ten tính hướng cao, và các thuộc tính bắt đầu của kênh tầng đối lưu được dùng bởi vi ba LOS. ăng ten càng lớn thì càng ít thành phần đa đường vì khi ăng ten lớn có nghĩa là trường nhìn thấy của ăng ten được giới hạn bởi các góc tới rất nhỏ. Ngoài ra, vì ăng ten là cố định, nên các thay đổi theo thời gian trong đặc tính kênh chỉ do các thay đổi điều kiện khí hậu gây ra. Các thay đổi này có thể coi là rất chậm khi so với các độ rộng băng tần. Vì vậy, mô hình Rummler là mô hình đa đường với pha đinh *rất chậm*.

Mô hình thứ hai, là các kênh vô tuyến di động. Các kênh này có nhiều thành phần đa đường do sử dụng các ăng ten vô hướng, nó bắt nhiều phản xạ với các trễ truyền sóng thay đổi lớn, đặc biệt trong các vùng thành phố. Chúng phải chịu pha đinh nhanh hơn, do số lượng lớn các thành phần đa đường phải chựu dịch pha sóng mang lớn trên các thay đổi khoảng cách nhỏ vì vậy có thể kết hợp mạnh hoặc yếu trên các khoảng cách nhỏ.

Mô hình Rummler cho các kênh vi ba mặt đất LOS

Một trong các mô hình được dùng rộng rãi nhất cho các tuyến vi ba mặt đất hoạt động trên dải tần số 4-6 GHz giữa các tháp cố định, được triển khai bởi Rummler. Mô hình này dựa trên một tập các giả định và các số liệu đo đạc được sử dụng để đạt được các giá trị bằng số của các tham số mô hình. Cho trước địa hình của tuyến và các tham số ăng ten, Rummler đưa ra giả thiết một mô hình 3 tia dạng:

$$y(t) = x(t) + \alpha . x(t-\tau_1) + \beta . x(t-\tau_2) \tag{14.76}$$

Trong đó *x(t)* và *y(t)* là các đầu vào/ra thông dải tương ứng. Ở dạng đường bao phức, mô hình đó có dạng:

$$\tilde{y}(t) = \tilde{x}(t) + \alpha \, e^{-j2\pi f_c \tau_1} \, \tilde{x}(t-\tau_1) + \beta \, e^{-j2\pi f_c \tau_2} \, \tilde{x}(t-\tau_2) \tag{14.77}$$

Và hàm truyền đạt tương đương thông thấp của kênh Rummler được cho bởi:

$$H(f) = 1 + \alpha \, e^{-j2\pi(f_c - f)\tau_1} + \beta \, e^{-j2\pi(f_c - f)\tau_2} \tag{14.78}$$

Đơn giản hóa đầu tiên của mô hình dựa trên giả thiết, trên độ rộng băng tần xét, thì $(f_c - f)\tau_1 << 1$, vì vậy $\exp(-j2\pi(f_c - f)\tau_1) \approx 1$ và:

$$H(f) \approx 1 + \alpha + \beta \, e^{-j2\pi(f_c - f)\tau_2} \tag{14.79}$$

Đơn giản hóa tiếp theo là giả sử rằng tần số "*hình V*", trong đó độ lớn của đáp ứng là cực tiểu tại $f_c + f_0$ trong trường hợp thông băng và tại f_0 trong mô hình thông thấp, vì vậy dạng cuối cùng của hàm truyền đạt tương đương thông thấp được viết là:

$$H(f) \approx a\left[1 - b . e^{-j2\pi(f_0 - f)\tau_2}\right] \tag{14.80}$$

Trong đó $a = 1 + \alpha$ là toàn bộ suy hao, $b = -\beta/(1+\alpha)$ là tham số định dạng. Giá trị của tham số số trễ τ_2 được chọn để phù hợp với số liệu đo, có giá trị $\tau_2 = \tau = 6.3ns$. Lưu ý rằng, đây là trễ nhỏ.

Đáp ứng biên độ của mô hình Rummler là:

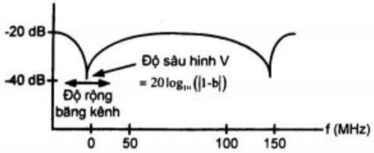
$$|H(f)|^2 = a^2\left[1 + b^2 - 2b.\cos(2\pi(f - f_0)\tau)\right] \tag{14.81}$$

Và một ví dụ của đáp ứng độ lớn được cho ở hình 14.18.

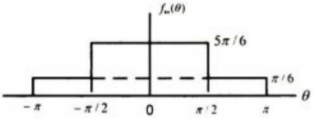
Các tham số *a* và *b* được chuẩn hóa và được biểu diễn đơn vị dB như minh họa trong bảng 14.3. Phân tích dữ liệu kênh nhận được các phân bố mũ cho B_1 và B_2 có giá trị trung bình là 3,8 dB. Tương tự như vậy, A_1 và A_2 là Gausơ với độ lệch chuẩn là 5 dB. Các trung bình là:

$$\mu = 24{,}6.\left(\frac{B^4 + 500}{B^4 + 800}\right)$$

Trong đó $B = B_1$ đối với A_1 và $B = B_2$ đối với A_2. Hàm mật độ xác suất của $\theta = 2\pi f_0 \tau$ được cho ở hình 14.19.



Hình 14.18: Minh họa đáp ứng biên độ của kênh Rummler



Hình 14.19: Hàm mật độ xác suất của θ

Bảng 14.3: Các tham số của mô hình Rummler

Trường hợp pha cực tiểu (b < 1)	Trường hợp pha không cực tiểu(b > 1)
$A_1 = 20\log_{10}(a)$	$A_2 = -20\log_{10}(ab)$
$B_1 = 20\log_{10}(1-b)$	$B_2 = 20\log_{10}(1-1/b)$

Để mô phỏng chớp ảnh của kênh Rummler, ta rút ra một tập các số ngẫu nhiên dưới đây.

1. Lấy ra một số U phân bố đều trong [0,1]. Nếu $U > 0,5$, thì coi là pha tối thiểu. Nếu $U < 0,5$, thì coi pha là không tối thiểu. (các thăng giáng pha tối thiểu và pha không tối thiểu được coi là đồng khả năng).

2. Rút ra một số ngẫu nhiên phân bố theo luật số mũ cho B_1 hoặc B_2.

3. Rút ra một số ngẫu nhiên Gausơ cho A_1 hoặc A_2 sử dụng giá trị của B_1 hoặc B_2.

4. Rút ra một số ngẫu nhiên cho θ và một tập tần số hình V tại $f_0 = \theta / 2\pi\tau$, $\tau = 6.3ns$.

Các tham số này định nghĩa một chớp ảnh của kênh Rummler. Vì kênh được coi là thay đổi rất chậm so với tốc độ ký hiệu, nên chuỗi các chớp ảnh của kênh là đầy đủ để ước lượng hiệu năng sử dụng mô phỏng Monte Carlo cho mỗi chớp ảnh được tạo ra bởi mô hình đó.

Lưu ý rằng: Các chương trình mô phỏng cho các kênh này được cho bởi **NVD14MFC1.m** và **NVD14MFC2.m** có trong phụ lục 14A.

Các mô hình cho các kênh di động

Các mô kênh rời rạc được dùng rộng rãi cho các kênh vô tuyến trong nhà và ngoài trời. Nhiều mô hình được dựa trên các số liệu được tập hợp trên phạm vi rộng môi trường truyền sóng. Một số lượng lớn mô hình toán học và mô hình được đề xuất gần đây, các nhà thiết kế hệ thống phải đối mặt với vấn đề chọn một tập trình bày các mô hình kênh để thể hiện các kênh, hệ thống hoạt động thỏa đáng. May thay, một vài hướng dẫn về việc chọn mô hình được cung cấp bởi nhóm các tiêu chuẩn quốc tế, định rõ một tập kênh "*miêu tả*" để phân tích và mô phỏng hiệu năng của các loại hệ thống truyền thông khác nhau. Dưới đây trình bày hai trường hợp điển hình.

Các mô hình kênh rời rạc cho các ứng dụng GSM

Hệ thống thông tin di động toàn cầu GSM là một tiêu chuẩn để truyền thông đi động trong băng tần từ (1 đến 2) GHz và sử dụng các kênh tần số vô tuyến RF 200 KHz để truyền thông, ghép kênh phân chia theo thời gian. Thời gian ký hiệu trong GSM là vài μs .

Các mô hình GSM được khuyến nghị là các mô hình rời rạc gồm 12 tia (đường) và được định rõ cho ba *kịch bản* khác nhau: *nông thôn, đồi núi và thành thị*. Với mỗi kịch bản, định rõ hai mô hình. Ngoài các mô hình 12 tia, tập các mô hình đơn giản hơn có 6 tia (đường) cũng được định nghĩa. Các mô hình 12 tia và 6 tia cho các vùng thành thị được cho ở bảng 14.4 và bảng 14.5. Ngoài ra, cũng có một mô hình cụ thể để kiểm tra hiệu năng bộ cân bằng Viterbi được dùng trong các hệ thống GSM và được cho trong bảng 14.6. Tất cả các công suất tương đối đơn vị dB và (1) và (2) định rõ hai mô hình tương đương.

Bảng 14.4: Lý lịch điển hình cho các vùng thành thị (mô hình 12 tia)

Tia	Thời gian tương đối giây	Thời gian tương đối giây	Công suất trung bình dB	Công suất trung bình dB	Phổ Doppler
	(1)	(2)	(1)	(2)	
1	0,0	0,0	-4,0	-4,0	Jakes
2	0,1	0,2	-3,0	-3,0	Jakes
3	0,3	0,4	0,0	0,0	Jakes
4	0,5	0,6	-2,6	-2,0	Jakes
5	0,8	0,8	-3,0	-3,0	Jakes
6	1,1	1,2	-5,0	-5,0	Jakes
7	1,3	1,4	-7,0	-7,0	Jakes
8	1,7	1,8	-5,0	-5,0	Jakes
9	2,3	2,4	-6,5	-6,0	Jakes
10	3,1	3,0	-8,6	-9,0	Jakes
11	3,2	3,2	-11,0	-11,0	Jakes
12	5,0	5,0	-10,0	-10,0	Jakes

Bảng 14.5: Lý lịch được rút gọn cho các khu vực thành thị (mô hình 6 tia)

Tia	Thời gian tương đối giây	Thời gian tương đối giây	Công suất trung bình dB	Công suất trung bình dB	Phổ Doppler
	(1)	(2)	(1)	(2)	
1	0,0s	0,0s	-3,0	-3,0	Jakes
2	0,2	0,2	0,5	0,0	Jakes
3	0,5	0,6	-2,0	-2,0	Jakes
4	1,6	1,6	-6,0	-6,0	Jakes
5	2,3	2,4	-8,0	-8,0	Jakes
6	5,0	5,0	-10,0	-10,0	Jakes

Cần lưu ý rằng thời gian ký hiệu trong hệ thống là vài μs và một vài trễ vi phân là $0.1\mu s$ nghĩa là nên dùng tốc độ lấy mẫu là 10M mẫu/giây để biểu diễn các trễ nhỏ này. Một giải pháp

khác, như đã được phác thảo trong phần trước, sử dụng TDL khoảng cách nhánh một ký hiệu (các hàm độ lợi nhánh tương quan) để giảm tải tính toán.

Bảng 14.6: Lý lịch để kiểm tra cân bằng

Tia	Thời gian tương đối	Công suất trung bình	Phổ doppler
1	0,0s	0,0dB	Jakes
2	3,2	0,0	Jakes
3	6,4	0,0	Jakes
4	9,6	0,0	Jakes
5	12,8	0,0	Jakes
6	16,0	0,0	Jakes

Các mô hình rời rạc cho các ứng dụng PCS

Đối với các hệ thống truyền thông PCS hoạt động trong băng tần 2GHz, các tiêu chuẩn vừa được chấp thuận trên một tập các mô hình rời rạc cho các môi trường điển hình. Các mô hình này được tóm tắt trong các bảng 14.7, 14.8, 14.9. Đối với mô hình được cho ở bảng 14.7, các cường độ tia $E\left\{|\tilde{a}|^2\right\}$ được cho ở bảng 14.8.

Bảng 14.7: Các tham số cho một mô hình ngoài trời 3 tia cho PCS

Môi trường	τ_1 (ns)	τ_2 (ns)	τ_3 (ns)	Doppler	Độ rộng băng tần Doppler
Đi bộ	0	1,500	14,500	Phẳng	12 Hz
Mạch vòng vô tuyến	0	1,500	14,500	Gausơ	12 Hz
Xe cộ	0	1,500	15,500	Jakes	180 Hz

Bảng 14.8: Cường độ tia

Tia	Công suất (dB)
1	0
2	-3
3	-6

Bảng 14.9: Các tham số của mô hình trong nhà PCS

Môi trường	Khoảng cách nhánh	Số nhánh	Phổ Doppler	Độ rộng băng tần Doppler
Nhà riêng	50 (ns)	2	Gausơ	3 Hz
Cơ quan	50	4	Gausơ	3
Thương mại	50	12	Phẳng	30

Nên lưu ý rằng, các hiệu số trễ là 50 ns (mô hình trong nhà) là rất nhỏ so với thời gian ký hiệu của các hệ thống PCS được đề xuất. Vì vậy, hoặc nên coi pha đinh là không chọn lọc tần số hoặc nên dùng mô hình TDL hạn chế băng tần có khoảng cách nhánh 1 ký hiệu cho mô phỏng.

Các mô hình kênh đa đường rời rạc cho các hệ thống CDMA băng rộng 3G

Hệ thống 3G sử dụng CDMA băng rộng hoạt động quanh 2 GHz. Các ví dụ về các mô hình kênh rời rạc được đề xuất cho các hệ thống 3G được cho ở bảng 14.10 (Trường hợp 1: trong nhà, Trường hợp 2: Trong nhà hoặc đi bộ, Trường hợp 3: xe cộ).

Bảng 14.10: Các tham số cho kênh W-CDMA 3G

Trường hợp 1 (3 km/h)		Trường hợp 2 (3 km/h)		Trường hợp 3 (120 km/h)	
Trễ (ns)	Công suất (dB)	Trễ (ns)	Công suất (dB)	Trễ (ns)	Công suất (dB)
0	0,0	0	0,0	0	0,0
244	-9,6	244	-12,5	244	-2,4
488	-35,5	488	-24,7	488	-6,5
				732	-9,4
				936	-12,7
				1220	-13,3
				1708	-15,4
				1953	-25,4

14.7.4. Mô hình cho kênh vô tuyến trong nhà

Các đặc tính pha đinh của các kênh vô tuyến trong nhà rất khác so với các kênh xe cộ do sự khác nhau về môi trường vật lý (kích thước, vật liệu, v.v...) và cơ chế truyền lan. Môi trường xe cộ ngoài trời được đặc tính hóa bởi các tế bào rộng (vài km) và ít thành phần đa đường hơn. Mặt khác, các môi trường trong nhà được đặc trưng hóa bởi các kích thước nhỏ hơn (vài chục m) và nhiều thành phần đa đường hơn vì nhiều phản xạ từ tường, bàn và các bề mặt khác. Một vài mô hình thống kê cho kênh trong nhà được rút ra từ đo đạc, các mô hình trong nhà có thể được phân loại như các mô hình đa đường rời rạc dày đặc có trải trễ rms trong khoảng 30 đến 300 ns với mỗi thành phần có các thống kê đường bao Ricean. Chỉ số tổn hao đường điển hình thay đổi từ 1,8 đến 4. Các kỹ thuật mô phỏng cho kênh trong nhà tương tự như các kênh đa đường khác. Tuy nhiên, gặp phải hiệu số trễ nhỏ trong môi trường trong nhà có thể cần phải chuyển các mô hình TDL khoảng các không đồng đều thành mô hình TDL khoảng cách đồng đều như đã đề cập ở phần 14.7.2.

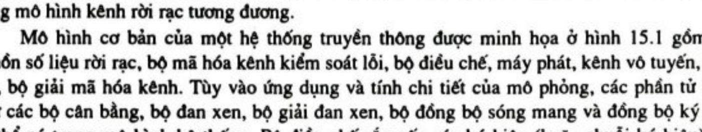
Chương 15

MÔ HÌNH KÊNH RỜI RẠC

15.1. Mở đầu

Trong chương này, ta xét các giải pháp khác cho bài toán lập mô hình kênh nhằm có được mô hình mô phỏng hiệu quả. Trong phần trước, kênh đã được định nghĩa ở dạng tạp âm, nhiễu và các nhiễu loạn khác, chúng kết hợp với tín hiệu phát tạo ra dạng sóng tạp âm và méo tại đầu vào máy thu. Tín hiệu phát, tập âm, nhiễu và các nhiễu loạn kênh khác, tất cả đều được biểu diễn bởi các mẫu của các dạng sóng. Kết quả là mô phỏng mức dạng sóng, số liệu được xử lý trên cơ sở *từng mẫu*. Tại đây, ta xét việc phân chia hóa hệ thống theo hướng loại bỏ nhiều phần được dùng trong mô phỏng mức dạng sóng. Kết quả là ta được mô hình kênh rời rạc trong đó việc mô phỏng trên cơ sở *từng ký hiệu*. *Động cơ* để thay thế mô hình kênh dạng sóng bằng mô hình kênh rời rạc là *tốc độ mô phỏng*. Ta sẽ thấy rằng, mô hình kênh rời rạc là sự trừu tượng hóa của kênh vật lý (dạng sóng), trong đó kênh được đặc tính hóa đầy đủ bằng *một tập nhỏ* các tham số. Xác định rõ các tham số này góp phần quan trọng vào quá trình mô hình hóa và phải được thực hiện thông qua đo đạc trên kênh vật lý, hoặc thông qua việc dùng một mô hình mức dạng sóng. Một khi có được mô hình khả tin, thì mô hình kênh dạng sóng có thể được thay bằng mô hình kênh rời rạc tương đương.

Mô hình cơ bản của một hệ thống truyền thông được minh họa ở hình 15.1 gồm có: nguồn số liệu rời rạc, bộ mã hóa kênh kiểm soát lỗi, bộ điều chế, máy phát, kênh vô tuyến, máy thu, bộ giải mã hóa kênh. Tùy vào ứng dụng và tính chi tiết của mô phỏng, các phần tử khác như các bộ cân bằng, bộ đan xen, bộ giải đan xen, bộ đồng bộ sóng mang và đồng bộ ký hiệu có thể có trong mô hình hệ thống. Bộ điều chế sắp xếp các ký hiệu (hoặc chuỗi ký hiệu) đầu vào thành một dạng sóng đầu ra. Dạng sóng phải chịu nhiều ảnh hưởng trong kênh. Các ảnh hưởng phải tính đến là tạp âm, giới hạn băng tần, nhiễu và pha đinh, chúng đều được đặc trưng hóa bằng các dạng sóng. Đầu vào máy thu cũng là một dạng sóng, là dạng sóng phát kết hợp với các ảnh hưởng xấu của kênh. Vai trò của máy thu là quan sát sóng đầu vào trong khoảng thời gian một ký hiệu hoặc một chuỗi các ký hiệu, và xác định chuỗi ký hiệu phát. Kênh trong trường hợp này được coi là kênh dạng sóng, vì đầu vào/ra của kênh đều được đặc tính là các dạng sóng. Đặc biệt lưu ý rằng, tất cả các thành phần hệ thống đều được đặc tính bởi các ánh xạ (sắp xếp) tất định, ngoại trừ kênh. Kênh thực hiện sắp xếp ngẫu nhiên (hay stochastic) tín hiệu vào thành tín hiệu ra của kênh.



Hình 15.1: Hệ thống truyền thông với mô hình kênh rời rạc

Thuật ngữ "*mô hình kênh rời rạc*" DCM được dùng để bao hàm tất cả các phần tử của hệ thống truyền thông nằm giữa hai điểm A và B trong đó đầu vào tại A là một véc tơ các ký hiệu

rời rạc (chuỗi đầu vào) $\mathbf{X}=[x_1,x_2,...,x_k,....]$ và đầu ra tại B là một véc tơ các ký hiệu rời rạc khác (chuỗi đầu ra) $\mathbf{Y}=[y_1,y_2,...,y_k,....]$.

Thường điểm A là đầu ra bộ mã hóa kênh (đầu vào bộ điều chế), điểm B là đầu vào bộ giải mã như được thấy ở hình 15.1. Lưu ý rằng, bằng cách phân chia hóa như hình 15.1, thì bộ điều chế, máy phát, kênh dạng sóng, máy thu đều là các bộ phận của mô hình kênh rời rạc. Quan hệ giữa các véc tơ vào/ra (X/Y) của kênh rời rạc đều bị ảnh hưởng bởi các thành phần hệ thống (như bộ lọc) và các nhiễu loạn ngẫu nhiên do kênh vật lý (kênh dạng sóng). Trong hệ thống truyền thông nhị phân dùng máy thu quyết định cứng, thì các phần tử của X và Y đều là các chuỗi nhị phân. Trong trường hợp máy thu quyết định mềm, các phần tử của Y sẽ từ bảng mẫu tự M-ary. Các lỗi truyền dẫn do tính không hoàn hảo của các phần tử hệ thống giữa điểm A và điểm B bao gồm kênh vật lý, chúng gây ra Y khác X.

Các mô hình kênh rời rạc mô tả cơ chế tạo lỗi có thể xảy ra. Các mô hình này được chia làm hai loại: *(i)* Lớp những mô hình kênh rời rạc "*không nhớ*", được dùng để mô hình hóa các lỗi truyền dẫn hoặc mô hình hóa các chuyển tiếp từ đầu vào kênh tới đầu ra kênh với giả định không có tương quan thời gian trong cơ chế chuyển tiếp (nghĩa là các xác suất chuyển tiếp vào/ra cho ký hiệu đầu vào kênh thứ *n* không bị ảnh hưởng bởi những gì xảy ra cho bất kỳ các ký hiệu đầu vào khác). Các mô hình này thường ứng dụng cho các kênh không có ISI hoặc pha đinh và tạp âm là AWGN. Trong hệ thống nhị phân quyết định cứng, giả định này có nghĩa là các lỗi bit không tương quan nhau. *(ii)* Lớp các mô hình kênh rời rạc "*có nhớ*", là lớp các mô hình đáng quan tâm hơn để ứng dụng vào tình huống ở đó việc chuyển tiếp các ký hiệu vào/ra bị *tương quan theo thời gian*. Trong trường hợp này, xác suất lỗi đối với ký hiệu thứ *n* phụ thuộc vào có xẩy ra lỗi ở lần truyền dẫn các ký hiệu trước đó hay không. Kênh pha đinh thường gặp phải trong truyền thông vô tuyến là minh họa rõ nét cho kênh có các *lỗi tương quan*. Các lỗi sẽ có xu hướng xuất hiện ở dạng các cụm khi kênh vô tuyến là kênh *pha đinh sâu* và những kênh này được coi là các kênh lỗi *cụm* hay kênh có *nhớ*.

Các mô hình kênh rời rạc là những mô hình xác suất có hiệu quả tính toán cao hơn so với các mô hình kênh dạng sóng. Hiệu quả được gia tăng nhờ hai nhân tố (tốc độ lấy mẫu và mức độ trừu tượng): *(i)* Tốc độ lấy mẫu, các mô hình kênh rời rạc được mô phỏng tại tốc độ ký hiệu trong khi đó các mô hình mức dạng sóng thường được mô phỏng tại *tốc độ* từ 8 đến 16 lần tốc độ ký hiệu. Điều này làm giảm đáng kể gánh nặng tính toán. *(ii)* Mức độ trừu tượng, trong khi mỗi khối riêng rẽ được mô phỏng chi tiết trong mô hình mức dạng sóng thì mô hình kênh rời rạc được *trừu tượng* ở mức cao hơn. Mức trừu tượng này sẽ làm giảm tải tính toán. Hai nhân tố này góp phần giảm thời gian mô phỏng.

Trong các trường hợp đơn giản, có thể rút ra được các mô hình kênh rời rạc theo phép giải tích từ các mô hình của các thành phần cơ bản giữa đầu vào A và đầu ra B của kênh rời rạc. Tuy nhiên, hầu hết các trường hợp, các mô hình kênh rời rạc được rút ra từ các mẫu lỗi đo hoặc mô phỏng giữa điểm A và điểm B. Các mô hình kênh rời rạc được dùng để thiết kế và phân tích ảnh hưởng của các thành phần *bên ngoài* phần hệ thống giữa điểm A và B. Ví dụ, các bộ mã hóa kiểm soát lỗi, bộ lập mã nguồn và bộ đan xen.

Mô mô hình hóa các kênh rời rạc *không nhớ* là quá trình tương đối dễ dàng. Ví dụ, trường hợp kênh đối xứng rời rạc không nhớ có đầu vào/ra nhị phân, tất cả cần phải đặc tính

hóa kênh là một con số, cụ thể là xác suất lỗi bit. Việc mô phỏng kênh bao gồm rút ra một con số ngẫu nhiên và so sánh với ngưỡng để quyết định bit truyền qua kênh có bị lỗi hay không. Theo đó, để mô phỏng quá trình truyền dẫn hàng triệu bit qua kênh, ta phải tạo ra hàng triệu con số ngẫu nhiên phân bố đều độc lập và so sánh với ngưỡng. Vì vậy, toàn bộ hệ thống được biểu diễn bởi kênh này được mô phỏng một cách hiệu quả. (So sánh mô phỏng này với mô phỏng mức dạng sóng - bao gồm việc tạo ra nhiều triệu mẫu để biểu diễn các dạng sóng và xử lý các mẫu này thông qua tất cả các khối chức năng của hệ thống).

Mô hình hóa các kênh rời rạc *có nhớ* là khó khăn hơn nhiều. Cơ chế tạo lỗi tương quan theo thời gian thường được mô hình hóa bằng một chuỗi Markov thời gian rời rạc, trong đó *mô hình trạng thái được dùng để đặc tính hóa các trạng thái khác nhau của kênh* và một *tập các xác suất truyền được dùng để bắt giữ tiến triển các trạng thái kênh*. Mỗi trạng thái được liên kết với một tập các xác suất truyền ký hiệu từ đầu vào tới đầu ra. Vì vậy, mô hình này phức tạp hơn và cần có nhiều tham số hơn so với kênh không tương quan. Cấu trúc mô hình và các giá trị của các tham số được ước tính từ các mẫu lỗi đo hoặc mô phỏng. Việc mô phỏng kênh rời rạc bao gồm tạo số ngẫu nhiên trước khi truyền dẫn mỗi ký hiệu nhằm xác định trạng thái kênh và sau đó rút ra một số ngẫu nhiên khác để xác định truyền từ đầu vào tới đầu ra. Trong khi thủ tục lập mô hình và ước tính tham số là phức tạp, thì việc mô phỏng cho mô hình Markov lại rất hiệu quả. Ta sẽ đề cập thêm những ý tưởng này trong các phần tiếp theo.

Theo đó, tiêu điểm của chương là triển khai các mô hình kênh rời rạc. Các mô hình kênh rời rạc là hấp dẫn vì dùng chúng giảm được rất nhiều độ phức tạp tính toán khi thực hiện mô phỏng. *Các tham số của mô hình có thể được xác định từ số liệu đo đạc hoặc từ kết quả mô phỏng mức dạng sóng.* Các mô hình kênh rời rạc được dùng rộng rãi trong các hệ thống truyền thông vô tuyến vì chúng được dùng để lập mô hình một cách hiệu quả các kênh pha đinh. Như trên đã đề cập, các mô hình kênh rời rạc là một trừu tượng hóa của các mô hình dạng sóng trong đó chúng đặc trưng hóa các đặc tính đầu vào/ra của kênh nhưng không lập mô hình chức năng vật lý của kênh. Thông qua tính trừu tượng này, gánh nặng tính toán được giảm.

Trước hết, ta xét cho mô hình hai trạng thái để thiết lập khái niệm ma trận chuyển dịch trạng thái, véc tơ phân bố trạng thái và ma trận tạo lỗi. Một vài kỹ thuật khác để xác định ma trận phân bố trạng thái bền được đề cập. Sau đó, các khái niệm này được mở rộng cho mô hình N trạng thái bao gồm việc triển khai các kỹ thuật để mô phỏng kênh một cách hiệu quả trên cơ sở mô hình kênh rời rạc.

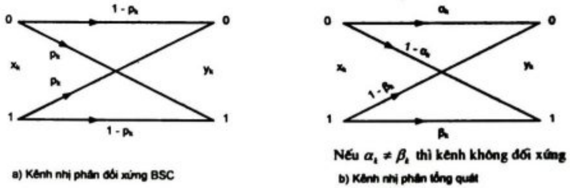
Tiếp đến, ta sẽ xét cho mô hình Gilbert 2 trạng thái và mô hình Fritchman N trạng thái như là các ví dụ của mô hình Markov. Với mục đích tính toán thì Mô hình Fritchman là hấp dẫn vì ma trận chuyển dịch trạng thái chứa một số lượng lớn số 0 (tương đối thưa). Mô hình Fritchman cũng cho thấy hiệu quả trong việc mô hình hóa các môi trường kênh pha đinh.

Trọng tâm của chương là xử lý triển khai các mô hình kênh rời rạc trên cơ sở dữ liệu đo đạc hoặc dữ liệu từ mô phỏng mức dạng sóng. Công cụ cơ bản được dùng để ước tính tham số là thuật toán Baum-Welch. Việc triển khai thuật toán Baum-Welch được phác thảo ngắn gọn và việc triển khai mô phỏng Matlab cho thuật toán Baum-Welch cũng được trình bày. Để cập việc lấy tỉ lệ ngăn chặn mất giá trị, là một mô hình Markov tương đương khối, mô hình này có thể được ước tính hiệu quả hơn mô hình Markov tổng quát.

Kết thúc chương bằng ba một ví dụ và hai trường hợp điển hình tóm tắt các kỹ thuật để ước tính các mô hình Markov và các mô hình Markov tương đương khối. Ví dụ 15.3 tập trung vào việc tạo một véc tơ lỗi được cho mô hình Markov. Tính hợp lý của kết quả dựa trên xác suất chiếm giữ trạng thái và xác suất lỗi. Trường hợp thứ nhất dùng thuật toán Baum-Welch để ước tính mô hình kênh. Tính hợp lệ gồm việc so sánh con số $\Pr\{0^m|1\}$ cho cả véc tơ lỗi gốc và véc tơ lỗi được tao ra bởi mô hình được ước tính. Trường hợp thứ 2 tương tự với trường hợp thứ nhất ngoại trừ mô hình Markov khối tương được dùng. Ví dụ 15.3 và hai trường hợp điển hình này có tác dụng như là một hướng dẫn sử dụng các công cụ được triển khai trong chương này.

15.2. Mô hình kênh rời rạc không nhớ

Trong kênh rời rạc *không nhớ*, việc sắp xếp từ đầu vào tới đầu ra là *tức thì* và được mô tả bằng một tập các xác suất truyền. Mô hình kênh rời rạc không nhớ đơn giản nhất là mô hình kênh nhị phân đối xứng (BSC), được minh họa ở hình 15.2(a). Đầu vào của kênh rời rạc là một chuỗi ký hiệu nhị phân, được ký hiệu là véc tơ X. Thành phần thứ *k* của véc tơ này là x_k, tương ứng với đầu vào kênh thứ *k*, và p_k tương ứng với xác suất lỗi truyền dẫn ký hiệu thứ *k*. Đối với kênh không nhớ, p_k không phụ thuộc vào *k*, vì vậy lỗi trên tất cả các ký hiệu đều bị ảnh hưởng bởi kênh với cùng một kiểu. (ở dạng tổng quát, nếu kênh có nhớ thì p_k là một hàm của p_{k-1}). Chuỗi *M* ký hiệu được xử lý thông qua kênh không nhớ bằng cách gọi mô hình kênh *M* lần liên tiếp. Với ký hiệu thứ *k*, đầu vào nhị phân là "0" thu đúng là "0" với xác suất là 1- p_k, và thu sai là "1" với xác suất lỗi là p_k. Đầu ra kênh thứ *k* được ký hiệu là y_k và chuỗi M ký hiệu của các đầu ra được ký hiệu là véc tơ Y.



Hình 15.2: Các mô hình kênh nhị phân

Mô hình kênh nhị phân đối xứng

Kênh là *đối xứng* trong đó các số 0 và số 1 đều bị ảnh hưởng bởi kênh theo cùng một kiểu. Lưu ý rằng kết quả của tính đối xứng là xác suất lỗi không phụ thuộc vào ký hiệu phát, vì vậy nguồn lỗi có thể được mô phỏng riêng biệt với nguồn thông tin (số liệu).

Đối với kênh nhị phân, quan hệ đầu vào/ra có thể được biểu diễn như sau:

$$\mathbf{Y} = \mathbf{X} \oplus \mathbf{E} \tag{15.1}$$

Trong đó X và Y là các véc tơ dữ liệu vào/ra, ⊕ là phép toán XOR, E là véc tơ lỗi. Cụ thể, E = $\{e_1, e_2, e_3,...\}$ là véc tơ nhị phân hay chuỗi có phần tử $\{0,1\}$ trong đó $e_k = 0$ biểu thị phần tử thứ *k* của X, x_k được thu đúng ($y_k = x_k$), và $e_k = 1$ biểu thị phần tử thứ *k* của X, x_k được thu sai ($y_k \neq x_k$). Tham số xác định hiệu năng của mô hình kênh đối xứng nhị phân là xác suất

lỗi P_E, có thể dễ dàng ước tính từ đo đạc hoặc từ việc phỏng hệ thống ở mức dạng sóng. Lưu ý rằng, đối với hệ thống nhị phân, ước tính xác suất lỗi theo Monte Carlo là trọng số *Hamming* của véc tơ lỗi E được chia cho N (số lượng các phần tử trong véc tơ lỗi E).

Lưu ý rằng, một khi biết được p_k, thì có thể tạo chuỗi lỗi tương ứng với N ký hiệu phát bằng cách gọi N lần bộ tạo số ngẫu nhiên phân bố đều và so sánh với số ngẫu nhiên với ngưỡng p. Cụ thể là:

$$e_k = \begin{cases} 1, & U_k \le p_k \\ 0, & U_k > p_k \end{cases} \tag{15.2}$$

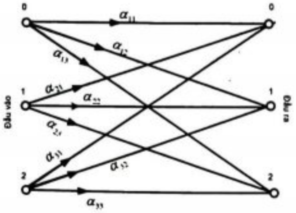
Trong đó U_k ký hiệu cho con số nhận được từ lần gọi thứ k bộ tạo số ngẫu nhiên phân bố đều. Nó quan trọng để hiểu rằng mô hình kênh rời rạc sẽ không tái tạo mẫu lỗi ban đầu, được dùng để xác định xác suất lỗi ban đầu p. Tuy nhiên, nó sẽ tái tạo một mẫu lỗi tương đương thống kê. Tính đơn giản của (15.2) giải thích lý do tại sao việc thay thế mô hình kênh rời rạc DCM cho phần hệ thống nằm giữa điểm A và B trong hình 15.1 dẫn đến hiệu quả tính toán.

Mô hình kênh nhị phân không đối xứng (tổng quát)

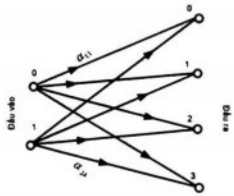
Mô hình phức tạp hơn một chút là *mô hình kênh nhị phân không đối xứng* trong đó xác suất lỗi có thể khác nhau đối với các số 0 và 1, như trong một số hệ thống thông tin quang. Điều này được minh họa ở hình 15.2(b). Mô hình này được đặc tính hóa bởi một tập bốn xác suất truyền từ đầu vào tới đầu ra, nhưng chỉ có hai trong số bốn xác suất truyền này α_k và β_k là độc lập nhau. Kênh là không đối xứng nếu $\alpha_k \neq \beta_k$.

Việc mở rộng mô hình này cho trường hợp đầu vào/ra thuộc về một M-ary ký tự của bảng mẫu tự là dễ dàng và được cho ở hình 15.3 với M = 3. Kênh được đặc tính hóa bằng một tập các xác suất truyền a_{ij}, $i, j = 0, 1, 2$ thể hiện xác suất có điều kiện:

$$\alpha_{ij} = \Pr\{\text{đầu ra} = j \mid \text{đầu vào} = i\} \tag{15.3}$$



Hình 15.3: Mô hình kênh rời rạc (3 đầu vào và 3 đầu ra)



Hình 15.4: Mô hình kênh rời rạc (hai đầu vào và 4 đầu ra)

Nếu tập các xác suất a_{ij} là hằng số, và vì vậy không phụ thuộc vào k, đầu ra kênh thứ k chỉ phụ thuộc vào đầu vào kênh thứ k. Với trường hợp này, mô hình ở hình 15.3 là không nhớ. Các xác suất truyền được ước tính bằng việc mô phỏng mức dạng sóng chi tiết trong đó đếm các truyền từ đầu vào thứ i tới đầu ra thứ j. Trong các mô hình này, các xác suất truyền được ước tính từ số liệu mô phỏng hoặc đo đạc là:

$$\alpha_{ij} = \frac{\text{Số lần ký hiệu đầu vào thứ } i \text{ truyền đến đầu ra thứ } j}{\text{Số lần xuất hiện ký hiệu đầu vào thứ } i} = \frac{n_{ij}}{N_i} \quad (15.4)$$

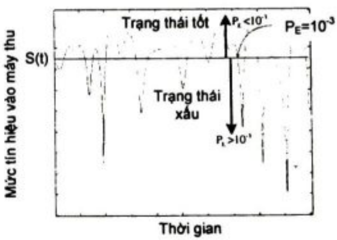
trong đó N_i là số lần xuất hiện ký hiệu đầu vào thứ i, n_{ij} là số lần ký hiệu đầu vào thứ i truyền tới đầu ra thứ j. Một ví dụ khác được cho ở hình 15.4, được dùng cho các kênh không nhớ trong đó đầu ra của kênh được lượng tử hóa thành nhiều mức từ các mức đầu vào, như trong giải mã quyết định mềm.

15.3. Mô hình Markov cho kênh rời rạc có nhớ

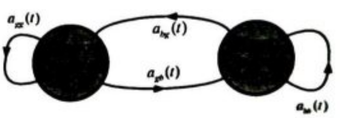
Với các kênh có nhớ, mô hình được dùng phổ biến nhất là mô hình Markov hữu hạn trạng thái rời rạc. Một vài lý do cho tính phổ biến của mô hình Markov. Các mô hình thuộc loại xử lý được theo phép giải tích (liên hệ với chương 1), lý thuyết của chúng dễ dàng được thiết lập theo lý thuyết thống kê, và chúng đã được ứng dụng thành công cho nhiều bài toán truyền thông quan trọng. Mô hình Markov đã được dùng để mô hình hóa đầu ra nguồn tin rời rạc như văn bản Anh ngữ (sự xuất hiện của chữ cái trong bảng mẫu tự Anh ngữ có thể được mô hình hóa như một chuỗi Markov). Tương tự, các giá trị mẫu của dạng sóng âm thanh hoặc hình ảnh cũng có thể được mô hình hóa bằng mô hình Markov (MM). Có thể trực tiếp ứng dụng các kỹ thuật mô hình hóa Markov để mô hình hóa và phân tích các kênh truyền thông rời rạc. Ngoài ra, các mô hình Markov cũng có thể được dùng để ước lượng dung lượng kênh rời rạc và thiết kế các kỹ thuật mã hóa kiểm soát lỗi tối ưu. Các mô hình Markov đã được dùng thành công để *đặc tính hóa* kênh pha đinh trong các hệ thống thông tin vô tuyến. Quan trọng nhất, các kỹ thuật tính toán hiệu quả là khả dụng để ước tính các tham số của các chuỗi Markov từ các mẫu lỗi đo đạc hoặc mô phỏng. Khi giới thiệu về hình Markov, trước hết ta xét mô hình hai trạng thái đơn giản. Sau đó, tổng quát hóa các kết quả cho mô hình N trạng thái.

15.3.1. Mô hình hai trạng thái

Để thiết lập tầng cho các mô hình kênh Markov, xét một kênh pha đinh trong đó cường độ tín hiệu thu trên mức ngưỡng hiệu năng chấp nhận được (khoảng thời gian khả dụng), và dưới mức ngưỡng (khoảng thời gian pha đinh sâu). Nếu ta chỉ quan tâm các điều kiện "ngưỡng trên" hoặc "ngưỡng dưới", thì ta có thể lập mô hình kênh vào một trong hai trạng thái như được thấy ở hình 15.5. Trong hình 15.5, ta có một trạng thái tốt g trong đó hiệu năng hệ thống chấp nhận được ($P_E < 10^{-3}$), và một trạng thái xấu b trong đó mức tín hiệu thu thấp đến mức làm cho xác suất lỗi không chấp nhận được ($P_E > 10^{-3}$).



Hình 15.5: Mức tín hiệu vào máy thu giả định và mô hình hai trạng thái



Hình 15.6: Mô hình Markov hai trạng thái

Vì vậy, trong mô hình hai trạng thái đơn giản được cho ở hình 15.5, các trạng thái có thể được thể hiện bằng tập:

$$S = \{g, b\} \tag{15.5}$$

Tiến triển theo thời gian, kênh chuyển từ trạng thái tốt sang trạng thái xấu và ngược lại. Tốc độ chuyển dịch và độ lâu của mỗi trạng thái phụ thuộc vào tương quan thời gian của quá trình pha đinh (thời gian nhất quán của kênh). Nếu thời gian được đo trong các *số gia của thời gian ký hiệu* (bit), thì ta có thể xây dựng mô hình kênh rời rạc như sau:

Tại thời điểm đầu của mỗi ký hiệu (bit), kênh thuộc một trong hai trạng thái. Nếu kênh ở trạng thái tốt, thì xác suất lỗi truyền dẫn là không đáng kể. Ngược lại, nếu kênh ở trạng thái xấu, thì xác suất lỗi truyền dẫn là lớn không thể chấp nhận được. Trước khi truyền một bit mới, kênh có thể chuyển trạng thái hoặc vẫn giữ nguyên trạng thái hiện tại. Việc chuyển đổi trạng thái diễn ra theo một tập các *xác suất chuyển dịch trạng* thái a_{ij}. Tồn tại các xác suất có điều kiện, và với mô hình hai trạng thái có bốn xác. Ký hiệu S_t và S_{t+1} là trạng thái của kênh tại thời điểm t và $t+1$ (*lưu ý rằng t & t+1 là chỉ số thời gian rời rạc*). Với việc ký hiệu này, ta định nghĩa 4 xác suất chuyển dịch trạng thái là:

$$\begin{aligned} a_{gg}(t) &= \Pr\{S_{t+1} = g \mid S_t = g\} \\ a_{gb}(t) &= \Pr\{S_{t+1} = b \mid S_t = g\} \\ a_{bg}(t) &= \Pr\{S_{t+1} = g \mid S_t = b\} \\ a_{bb}(t) &= \Pr\{S_{t+1} = b \mid S_t = b\} \end{aligned} \tag{15.6}$$

Phương trình trên được biểu diễn ở dạng ma trận chuyển dịch trạng thái sau:

$$\underbrace{A(t)}_{\text{Ma trận chuyển dịch trạng thái}} = \underbrace{\begin{bmatrix} a_{gg}(t) & a_{gb}(t) \\ a_{bg}(t) & a_{bb}(t) \end{bmatrix}}_{\text{Tổng các phần tử trên một hàng = 1}} \tag{15.7}$$

Lưu ý rằng, vì ta đang xét mô hình hai trạng thái, nên giả sử $S_t = g$ thì ta phải có $S_{t+1} = g$ (kênh vẫn ở trạng thái tốt) hoặc $S_{t+1} = b$ (kênh *chuyển sang* trạng thái xấu). Vì vậy, mỗi hàng của ma trận trạng thái phải có tổng bằng 1. Thể hiện ở dạng lưu đồ chuyển dịch trạng thái của mô hình hai trạng thái được cho ở hình 15.6.

Tiếp theo, giả sử mô hình kênh là *dừng*, theo đó ma trận chuyển dịch trạng thái $A(t)$ được cố định vì vậy $A(t) = A$. Tuy nhiên, nếu mô phỏng được thực hiện với phân bố xác suất trạng thái khởi đầu Π_0 khác với phân bố trạng thái *bền vững*, đôi khi cần có phân bố trạng thái để tiến đến giá trị trạng thái bền vững Π_{ss}. Cần phải lưu ý đến xác suất tìm kiếm mô hình trong trạng thái cho trước. Định nghĩa Π_t là phân bố xác suất trạng thái tại thời điểm t. Cụ thể là:

$$\underbrace{\Pi_t}_{\substack{\text{Phân bố xác suất} \\ \text{trạng thái tại thời điểm t}}} = \underbrace{\begin{bmatrix} \pi_{t,g} & \pi_{t,b} \end{bmatrix}}_{\substack{\text{Các xác suất tìm kiếm kênh} \\ \text{ở trạng thái tốt hoặc xấu} \\ \text{tại thời điểm t}}} \tag{15.8}$$

Trong đó $\pi_{t,g}$ và $\pi_{t,b}$ biểu diễn các xác suất tìm kiếm kênh trong trạng thái tốt hoặc xấu tại thời điểm *t*. Theo định nghĩa ma trận chuyển dịch trạng thái, phân bố trạng thái tại thời điểm *t + 1* được cho bởi:

$$\Pi_{t+1} = \Pi_t A \tag{15.9}$$

Theo cách tương tự, phân bố trạng thái tại thời điểm *t + 2* là:

$$\Pi_{t+2} = \Pi_{t+1} A = (\Pi_t A) A = \Pi_t A^2 \tag{15.10}$$

Vì vậy, ở dạng tổng quát:

$$\underbrace{\Pi_{t+k}}_{\text{Phân bố trạng thái tại thời điểm } t+k} = \underbrace{\Pi_t}_{\text{Phân bố trạng thái tại thời điểm } t} \cdot \underbrace{A^k}_{\text{Ma trận chuyển dịch trạng thái tại bước thứ } k} \tag{15.11}$$

Trong đó A^k thể hiện ma trận chuyển dịch trạng thái bước thứ *k*.

Hầu hết nhưng không phải là tất cả, các quá trình Markov chuyển dần về phân bố xác suất trạng thái *bền vững* theo sự tiến triển thời gian. Giả sử quá trình Markov hội tụ về phân bố giá trị trạng thái bền $\Pi_{t+k} = \Pi_t$ khi *t* đủ lớn và *k* bất kỳ, theo đó:

$$\Pi_{ss} = \Pi_{ss} A^k = \begin{bmatrix} \pi_g & \pi_b \end{bmatrix} \tag{15.12}$$

Với *k* bất kỳ. Dễ dàng nhận thấy rằng, khi giá trị k đủ lớn, thì các hàng của ma trận chuyển dịch trạng thái tại bước thứ *k* A^k tạo ra giá trị trạng thái bền vững Π_{ss}. Tính hội tụ của Π_0 về Π_{ss} được minh họa trong ví dụ dưới đây:

Ví dụ 15.1: Cho ma trận chuyển dịch trạng thái sau:

$$A = \begin{bmatrix} 0{,}98 & 0{,}02 \\ 0{,}05 & 0{,}95 \end{bmatrix} \tag{15.13}$$

Với phân bố trạng thái khởi đầu là:

$$\Pi_0 = [0{,}50 \quad 0{,}50] \tag{15.14}$$

Chương trình **Matlab NVD15_MMtransient.m** minh họa quá trình hội tụ của Π_0 về Π_{ss}.

Chạy chương trình nhận được kết quả:

Cac xac suat trang thai ben la: 0,71412 và 0,28588.

Gia tri cua ma tran chuyen dich trang thai A^N la:

ans =

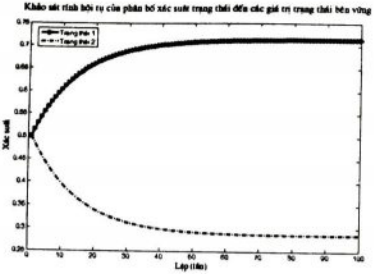
0,7145 0,2855

0,7138 0,2862

Thấy rõ, việc tính toán giá trị của Π_{ss} lặp lại dùng (15.9) và bằng cách tăng A tới một giá trị lớn để có được các kết quả phù hợp như mong muốn. Cách thức phân bố xác suất hội tụ đến giá trị trạng thái bền được thấy trên hình 15.7.

Trước khi rời khỏi mô hình hai trạng thái, cần phải đề cập ma trận tạo lỗi, được định nghĩa là:

$$B = \begin{bmatrix} \Pr\{C \mid g\} & \Pr\{C \mid b\} \\ \Pr\{E \mid g\} & \Pr\{E \mid b\} \end{bmatrix} \tag{15.15}$$



Hình 15.7: Quá trình hội tụ của phân bố xác suất trạng thái đến các giá trị trạng thái bền

Trong đó "*C*" và "*E*" biểu thị việc thực hiện quyết định đúng & sai tương ứng. Bằng phép nhân ma trận, theo đó xác suất quyết định đúng và sai không điều kiện P_C và P_E được cho bởi:

$$\begin{bmatrix} P_C & P_E \end{bmatrix} = \Pi_{SS} B^T \tag{15.16}$$

Trong đó Π_{ss} là ma trận phân bố trạng thái bền và B^T là chuyển vị của ma trận tạo lỗi *B*.

Lưu ý rằng, nếu tất cả các phần tử của B khác không, thì trạng thái có thể tạo ra một mỗi mặc dù các xác suất lỗi có thể khác khá nhiều đối với các trạng thái khác nhau. Vì vậy, khi quan sát lỗi, trạng thái tạo ra lỗi không thể nhận biết được. Đó là lý do mà ta gọi là mô hình "*ẩn*". Những mô hình này được cho là những mô hình Markov ẩn (HMM).

Ví dụ 15.2: Trong ví dụ này, mô phỏng kênh dùng mô hình Markov được sáng tỏ và thực hiện tính toán xác suất lỗi hệ thống. Giả sử các lỗi được tạo ra trong trạng thái, hiển nhiên xác suất lỗi trong trạng thái tốt $\Pr\{E \mid g\}$ sẽ nhỏ hơn nhiều so với xác suất lỗi trong trạng thái xấu $\Pr\{E \mid b\}$. Cụ thể, ta xác định các xác suất lỗi có điều kiện:

$$\Pr\{E \mid g\} = 0{,}0005 \tag{15.17}$$

$$\Pr\{E \mid b\} = 0{,}1000 \tag{15.18}$$

Với các xác suất lỗi này, ma trận tạo lỗi *B* có dạng:

$$B = \underbrace{\begin{bmatrix} 0{,}9995 & 0{,}9000 \\ 0{,}0005 & 0{,}1000 \end{bmatrix}}_{\text{Tổng các phần tử trên một cột} = 1} \tag{15.19}$$

Ngoài ra, chuỗi Markov được định nghĩa bởi ma trận chuyển dịch trạng thái:

$$A = \underbrace{\begin{bmatrix} 0,98 & 0,02 \\ 0,05 & 0,95 \end{bmatrix}}_{\text{Tổng các phần tử trên mỗi hàng = 1}} \tag{15.20}$$

Như trong ví dụ trước.

Chương trình Matlab để mô phỏng kênh được cho bởi **NVD15_hmm2.m** trong Phụ lục 5A.

Kết quả chạy chương trình mô phỏng **NVD15_hmm2.m** cho bài toán này là: $P_E = 0,0285$.

Vì vậy, ta có:

$$P_E = 0,0285 \tag{15.21}$$

Lưu ý rằng, P_E được tính toán theo kiểu này là biến ngẫu nhiên. Bây giờ ta "*kiểm tra-tính hợp lý*" kết quả mô phỏng này.

Từ ví dụ trước, các xác suất trạng thái bền vững là:

$$\Pi_{ss} = [0,7141 \quad 0,2859] \tag{15.22}$$

Từ (15.16) ta có:

$$[P_C \quad P_E] = [0,7141 \quad 0,2859]\begin{bmatrix} 0,9995 & 0,0005 \\ 0,9000 & 0,1000 \end{bmatrix} \tag{15.23}$$

Cho ta:

$$[P_C \quad P_E] = [0,9711 \quad 0,0289] \tag{15.24}$$

Nó $P_E = 0,0289$ rất sát với giá trị được cho bởi mô phỏng $P_E = 0,0285$. Vì vậy, kết quả mô phỏng là hợp lý.

Phiên bản tổng quát hơn của mô hình Markov hai trạng thái được xây dựng với một số lượng lớn các trạng thái, với nhiều trạng thái tốt và xấu với mức độ thay đổi về "*tính tốt*" hoặc "*tính xấu*" tương ứng với sự tăng dần mức độ pha đinh và/hoặc tạp âm. Kết quả là mô hình Markov N trạng thái, là sự tổng quát hóa của mô hình hai trạng thái.

15.3.2. Mô hình Markov N trạng thái

Một mô hình (quá trình) Markov N trạng thái cho kênh truyền thông rời rạc được định nghĩa bởi nhiều tham số. Tập các trạng thái S được biểu thị bằng:

$$S = \{1, 2, 3, ..., N\} \tag{15.25}$$

Và giống như trên, S_t biểu thị cho trạng thái tại thời điểm t. Vì vậy, S_t đi khắp tập S. Ta quan tâm xác suất $\Pi_{t,i}$ là xác suất mà mô hình Markov sẽ ở trạng thái i tại thời điểm t, nghĩa là:

$$\Pi_{t,i} = \Pr[S_t = i], \qquad i = 1,2,...,N \tag{15.26}$$

Các xác suất chuyển dịch trạng thái a_{ij} biểu thị xác suất chuyển từ trạng thái i tại thời điểm t ($S_t = i$) sang trạng thái j tại thời điểm $t + 1$ ($S_{t+1} = j$). Ở dạng phương trình:

$$a_{ij} = \underbrace{\Pr\left[S_{t+1} = j \mid S_t = i\right]}_{\substack{\text{Xác suất chuyển dịch trạng thái: từ trạng thái} \\ i \text{ ở thời điểm } t \text{ sang trạng thái } j \text{ ở thời điểm } t+1}} \quad , \quad i, j = 1, 2, ..., N \tag{15.27}$$

Các chuyển dịch này thường diễn ra trong một số gia thời gian bằng thời gian một bit hoặc một ký hiệu. Cuối cùng, ta có các xác suất (ký hiệu tin bị lỗi) chuyển dịch từ đầu vào tới đầu ra. Các ký hiệu lỗi đối với bảng mẫu tự ký hiệu *M-ary* được biểu thị bởi tập:

$$E = \{e_1, e_2, ..., e_M\} \tag{15.28}$$

Với trường hợp nhị phân quen thuộc thì:

$$E = \{0, 1\} \tag{15.29}$$

Trong đó 0 biểu thị không lỗi và 1 biểu thị lỗi. Xác suất xuất hiện ký hiệu lỗi e_k với giả thiết mô hình đang ở trạng thái *i* được biểu thị là $b_i(e_k)$. Ở dạng phương trình ta có:

$$\underbrace{b_i(e_k) = \Pr\left[e_k \mid S_t = i\right]}_{\substack{\text{Xác suất xảy ra ký hiệu lỗi thứ } k,\ (e_k), \\ \text{với giả thiết mô hình đang} \\ \text{ở trạng thái thứ } i \text{ tại thời điểm } t}} \tag{15.30}$$

Trong các ứng dụng quyết định cứng nhị phân, thì các tham số $b_i(e_k)$ được biểu diễn bởi một ma trận 2 hàng và *N* cột, trong đó *N thể hiện số trạng thái trong mô hình*. Vì vậy, ma trận tạo lỗi *B* có dạng:

$$B = \underbrace{\underbrace{\begin{bmatrix} \overbrace{\begin{matrix} b_{11} & b_{12} & \cdots & b_{1i} & \cdots & b_{1N} \end{matrix}}^{\text{Các xác suất quyết định đúng}} \\ \underbrace{\begin{matrix} b_{21} & b_{22} & \cdots & b_{2i} & \cdots & b_{2N} \end{matrix}}_{\text{Các xác suất lỗi}} \end{bmatrix}}_{\text{Tổng các phần tử trên mỗi cột} = 1}}_{\substack{\text{Trường hợp không phải là nhị phân hoặc quyết định mềm} \\ \text{thì ma trận tạo lỗi B có nhiều hơn 2 hàng}}} \tag{15.31}$$

Trong đó b_{1i} biểu thị xác suất quyết định đúng với điều kiện mô hình ở trạng thái *i*, và b_{2i} thể hiện xác suất lỗi với điều kiện hệ thống ở trạng thái *i*. Với phương trình này, lưu ý rằng các cột của *B* phải có tổng bằng 1. Với các ứng dụng phi nhị phân hoặc các ứng dụng *quyết định mềm* nhị phân, thì ma trận tạo lỗi B sẽ có *nhiều hơn hai hàng*.

Những tham số này định nghĩa quá trình (mô hình) Markov rời rạc theo thời gian hoạt động tại tốc độ chuyển dịch bằng tốc độ ký hiệu của hệ thống truyền thông, và đầu ra của quá trình bao gồm *hai chuỗi*: chuỗi các trạng thái $\{S_t\}$ và chuỗi các ký hiệu lỗi $\{E_t\}$, trong đó *t* là *chỉ số thời gian* được đánh chỉ số trên tập số nguyên $\{0, 1, 2....\}$. Thông thường, chỉ có đầu vào và đầu ra của kênh và vì thế chuỗi lỗi có thể quan sát được. Bản thân chuỗi trạng thái không thể quan sát được. Vì vậy, chuỗi trạng thái là **"*ẩn*"** hoặc không nhìn thấy từ các quan sát bên ngoài và mô hình Markov được gọi là mô hình Markov *ẩn*.

15.3.3. Quá trình Markov bậc một

Quá trình Markov bậc *m* (nhớ *m*), được định nghĩa là xác suất của trạng thái tại thời điểm $t + 1$, S_{t+1} được cho bởi:

$$\Pr\left[S_{t+1}\middle|S_t,S_{t-1},\dots\right]=\Pr\left[S_{t+1}\middle|S_t,S_{t-1},\dots,\underbrace{S_{t-m-1}}\right] \quad (15.32)$$

Cho thấy xác suất phụ thuộc vào m trạng thái trước đó.

Với quá trình Markov bậc nhất, thì xác suất của trạng thái tại thời điểm t + 1 chỉ là một hàm hàm của trạng thái *trước đó*. Nói cách khác:

$$P_r\left[S_{t+1}\mid S_t,S_{t-1},\dots\right]=\Pr\left[S_{t+1}\mid S_t\right] \quad (15.33)$$

Đối với các mô hình kênh, ta sẽ *dùng một mô hình Markov bậc nhất*. Đối với việc mô hình hóa nguồn số liệu, có thể cần đến các mô hình bậc cao hơn. Ví dụ, mô hình hóa các chuỗi ký hiệu trong văn bản Anh ngữ có thể cần đến mô hình Markov bậc 2 hoặc bậc 3, vì xác suất xuất hiện của một chữ cái cho trước có thể phụ thuộc vào hai hoặc ba chữ cái trước đó.

15.3.4. Tính dừng

Vì tính dừng thường được giả định trong quá trình mô hình hóa các phần tương tự của kênh truyền thông, nên cũng thường giả định tính dừng của mô hình Markov đối với các kênh rời rạc. Tính dừng để đề cập các tham số của mô hình, nghĩa là các xác suất $\prod_{i,j}$, $a_{ij}(t)$ và $b_j(e_k)$ không phụ thuộc vào t. Trong trường hợp này, ta có:

$$\underbrace{\Pr\left[S_{t+1}=i\right]}_{\substack{\text{Xác suất mô hình ở}\\ \text{trạng thái } i\\ \text{tại thời điểm } t+1}}=\Pi_i=\sum_{k=1}^{N}\underbrace{\Pr\left[S_t=k\right]}_{\substack{\text{xác suất mô hình}\\ \text{ở trạng thái } k\\ \text{tại thời điểm } t\\ \text{(xác suất trạng thái } k)}}\cdot\underbrace{\Pr\left[S_{t+1}=i\mid S_t=k\right]}_{\substack{\text{xác suất mô hình chuyển từ}\\ \text{trạng thái } k \text{ tại ở thời điểm } t\\ \text{sang trạng thái } i \text{ tại thời điểm } t+1}} \quad (15.34)$$

$$=\sum_{k=1}^{N}\pi_k a_{ki},\qquad i=1,\dots,N$$

Trong đó các số hạng a_{ki} là các phần tử trong cột thứ i của *ma trận chuyển dịch trạng thái:*

$$A=\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1i} & \dots & a_{1N}\\ a_{21} & a_{22} & \dots & a_{2i} & \dots & a_{2N}\\ \dots & \dots & \ddots & \dots & \ddots & \dots\\ a_{N1} & a_{N2} & \dots & a_{Ni} & \dots & a_{NN}\end{bmatrix} \quad (15.35)$$

Và π_k là các phần tử của véc tơ phân bố xác suất trạng thái:

$$\Pi=\left[\pi_1\quad \pi_2\quad \dots\quad \pi_N\right] \quad (15.36)$$

Như trong trường hợp mô hình hai trạng thái, ma trận chuyển dịch *n tầng* được cho bởi A^n. Như đã giải thích trong ví dụ 15.1 đối với mô hình 2 trạng thái, (15.34) cùng với ràng buộc:

$$\sum_{i=1}^{N}\pi_i=1 \quad (15.37)$$

Hàm ý rằng các xác suất trạng thái π_i được xác định *duy nhất* từ các xác suất chuyển dịch trạng thái a_{ij}. *Vì vậy*, mô hình Markov hoàn toàn được định nghĩa bởi: *(i)* Ma trận xác suất

chuyển trạng thái A (các phần tử của A là các xác suất chuyển dịch trạng thái của mô hình); *(ii)* Ma trận tạo lỗi B (ma trận xác suất chuyển dịch ký hiệu từ đầu vào tới đầu ra B, các phần tử của B là các xác suất chuyển dịch ký hiệu từ đầu vào tới đầu ra):

$$B = \begin{bmatrix} b_{11} & \cdots & b_{1N} \\ \vdots & \ddots & \vdots \\ b_{M1} & \cdots & b_{MN} \end{bmatrix} \tag{15.38}$$

Như đã thảo luận trước.

15.3.5. Mô phỏng mô hình Markov

Một khi rút ra được mô hình Markov rời rạc, thì việc mô phỏng mô hình là khá dễ dàng. Thấy rõ ở ví dụ 15.2 cho mô hình 2 trạng thái, và kỹ thuật dùng trong ví dụ 15.2 giờ đây được tổng quát hóa cho mô hình N trạng thái.

Giả sử rằng mô hình đang ở trạng thái k tại thời điểm t, và cả trạng thái tiếp theo và thành phần của véc tơ lỗi e_{k+1} tương ứng đều được mô phỏng. Mô phỏng được tiến hành theo hai bước cơ bản sau:

Bước 1 *(xác định xác suất chuyển dịch trạng thái a_{ki} nghĩa là thiết lập trạng thái mới):* Trước hết là tạo hai biến ngẫu nhiên phân bố đều U_1 và U_2. Biến ngẫu nhiên thứ nhất U_1 được dùng để xác định trạng thái mới. Giả sử, mô hình đang ở trạng thái k, thì các xác suất chuyển dịch được xác định bởi hàng thứ k của ma trận chuyển dịch trạng thái A. Hàng thứ k được định rõ bởi véc tơ A_k:

$$A_k = \begin{bmatrix} a_{k1} & a_{k2} & \dots & a_{ki} & a_{k,i+1} & \dots & a_{kN} \end{bmatrix} \tag{15.39}$$

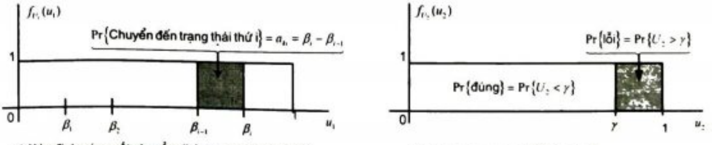
Các xác suất tích lũy liên kết với A_k được ký hiệu là β_k và được cho bởi:

$$\beta_k = \sum_{j=1}^{k} a_{kj} \tag{15.40}$$

Lưu ý rằng sẵn có hàm **cumsum** trong thư viện Matlab thực hiện xắp sếp véc tơ A_k vào một véc tơ có các phần tử β_k. Xác suất tạo sự chuyển dịch từ trạng thái k sang trạng thái i theo định nghĩa là a_{ki}, được cho bởi:

$$a_{ki} = \beta_i - \beta_{i-1} \tag{15.41}$$

Xác suất này được biểu diễn bởi vùng đậm trong hình 15.8(a).



a) Xác định xác suất chuyển dịch trạng thái (bước 1)

b) Xác định xác suất lỗi (bước 2)

Hình 15.8: Các bước mô phỏng mô hình Markov

Bước 2 *(xác định xác suất lỗi ở trạng thái mới)*: Sau khi trạng thái mới được thiết lập, bước tiếp theo là xác định lỗi có xẩy ra ở trạng thái mới không. Muốn vậy, ta so sánh biến ngẫu nhiên phân bố đều thứ hai U_2 với ngưỡng γ được cho ở hình 15.8(b). Xác suất quyết định đúng là $\Pr\{U_2 < \gamma\}$, và xác suất lỗi là $\Pr\{U_2 > \gamma\}$ là vùng đậm trong hình 15.8(b). Vì trạng thái hiện tại là i, nên giá trị của γ được cho bởi phần tử b_{1i} (hàng 1, cột i) trong ma trận quan sát trạng thái B.

Mã chương trình Matlab để thực hiện thủ tục mô phỏng cho bài toán này được cho bởi **NVD15_ext1.m** trong Phụ lục 15A.

Bây giờ ta sẽ giải thích quá trình này trong ví dụ dưới đây.

Ví dụ 15.3: *[Mô hình ba trạng thái quyết định cứng]:* Trong ví dụ này, một chuỗi nhị phân có chiều dài L = 20000 được tạo ra để thể hiện các lỗi ký hiệu trên kênh. Chuỗi lỗi này được tạo ra bằng cách dùng mô hình Markov ba trạng thái. Nếu cho trước một phần tử trong chuỗi đó là số "1" nhị phân, thì một lỗi ký hiệu kênh được ghi vào vị trí tương ứng với phần tử đó. Một số "0" nhị phân biểu thị rằng không xẩy ra lỗi tại vị trí đó.

Ta giả thiết rằng các lỗi có thể xẩy ra trong bất kỳ trạng thái nào thuộc ba trạng thái đó. Cụ thể, ta giả thiết:

$$\Pr\{E \mid S_1\} = 0{,}0010 \tag{15.42}$$

$$\Pr\{E \mid S_2\} = 0{,}0500 \tag{15.43}$$

$$\Pr\{E \mid S_3\} = 0{,}0100 \tag{15.44}$$

Tương ứng với ma trận xác suất lỗi (ma trận tạo lỗi) là:

$$B = \underbrace{\begin{bmatrix} \overbrace{0{,}9990 \quad 0{,}9500 \quad 0{,}9900}^{\text{Các xác suất quyết định đúng}} \\ \underbrace{\underbrace{0{,}0010 \quad 0{,}0500 \quad 0{,}0100}_{\text{Các xác suất quyết định sai}}}_{\text{Tổng các phần tử trên mỗi cột bằng 1}} \end{bmatrix}_{(\text{quyết định cứng }=2)\times(\text{số trạng thái }N=3)}}_{\substack{\text{Ma trận các xác suất (lỗi) chuyển dịch ký hiệu vào ra:} \\ \text{Số hàng của B thể hiện quyết định cứng hoặc mềm} \\ \text{Số cột của B thể hiện số trạng thái trong mô hình}}} \tag{15.45}$$

Ma trận chuyển dịch trạng thái đối với mô hình Markov ba trạng thái được giả thiết là:

$$A = \underbrace{\begin{bmatrix} 0{,}80 & 0{,}10 & 0{,}10 \\ 0{,}20 & 0{,}60 & 0{,}20 \\ 0{,}02 & 0{,}08 & 0{,}90 \end{bmatrix}_{3x3}}_{\sum_{j=1}^{3} a_{i,j}=1,\ (\text{tổng số phần tử trên hàng }=1)} \tag{15.46}$$

Ta giả sử mô hình này khởi đầu ở trạng thái 1.

Mã chương trình Matlab **NVD15_errvector.m** tạo chuỗi lỗi được cho ở Phụ lục 15A. Khi chạy chương trình Matlab **NVD15_errvector.m** phải nhập các tham số đầu vào là L, ma

trận B và A. Các giá trị mặc định là $L = 20000$ và các ma trận B và A lần lượt được định nghĩa bởi (15.45) và (15.46). Chương trình tạo ra hai đầu ra: *(i)* véc-tơ lỗi **out** cho biết các vị trí lỗi trong chuỗi của L lần truyền dẫn; *(ii)* véc tơ chuỗi trạng thái **state_seq** trình bày chi tiết các chuyển dịch trạng thái. Có thể tính toán: *(i)* xác suất tìm kiếm mô hình ở một trạng thái cho trước từ véc tơ chuỗi trạng thái; *(ii)* xác suất lỗi được mô phỏng từ véc tơ lỗi. Các tính toán này được thực hiện trong mã chương trình Matlab **NVD15rhmmtest.m**. Chạy chương trình này với các giá trị mặc định tạo ra các đầu ra dưới đây, lưu ý rằng các hội thoại (yêu cầu đầu vào của chương trình) liên quan đến việc loại bỏ hoặc chấp nhận giá trị mặc định được ẩn.

```
>> NVD15_hmmtest
    Các kết quả mô phỏng:
    Xác suất trạng thái 1 là:      0,2432
    Xác suất trạng thái 2 là:      0,1634
    Xác suất trạng thái 3 là:      0,5934
    Xác suất lỗi là:               0,01445
```

Lưu ý rằng, các giá trị mô phỏng được cho đối với các xác suất trạng thái và các xác suất lỗi là các biến ngẫu nhiên (giá trị ngẫu nhiên). Giảm phương sai của các biến ngẫu nhiên này bằng cách tăng độ dài véc tơ lỗi.

Có thể "*kiểm tra tính hợp lý*" đối với các giá trị trên bằng cách tính toán các xác suất trạng thái và xác suất lỗi trực tiếp từ hai ma trận B và A. Mã chương trình Matlab cùng với kết quả đầu ra như sau (lưu ý rằng, phải nhập các ma trận A và B trước khi thực hiện các dòng lệnh này):

>> A100=A^100

$$A100=\begin{bmatrix}0,2353 & 0,1765 & 0,5882\\ 0,2353 & 0,1765 & 0,5882\\ 0,2353 & 0,1765 & 0,5882\end{bmatrix}$$

>> $PE = B * A100'$

$$PE=\begin{bmatrix}0,9851 & 0,9851 & 0,9851\\ 0,0149 & 0,0149 & 0,0149\end{bmatrix}$$

Lưu ý rằng, xác suất trạng thái lý thuyết S_1, S_2, S_3 lần lượt là 0,2353; 0,1765; 0,5882 và xác suất lỗi lý thuyết là 0,0149. Thấy rõ, các kết quả này hơi khác so với các kết quả mô phỏng, như đã đề cập ở trên, là các biến (giá trị) ngẫu nhiên. Phương sai của các biến ngẫu nhiên này có thể được giảm như đã đề cập ở trên, để được tường minh ta nên kiểm tra bằng cách chọn giá trị L lớn hơn.

15.4. Mô hình Gillbert và Fritchman

Đặc tính hóa các kênh rời rạc dùng các chuỗi Markov hữu hạn trạng thái, rời rạc theo thời gian đã được đề xuất bởi Gillbert, Fritchman và nhiều tác giả khác.

Mô hình Gillbert là một mô hình hai trạng thái có một trạng thái tốt không lỗi, và một trạng thái xấu với một xác suất lỗi p. Mô hình này đã được Gillbert dùng để tính dung lượng

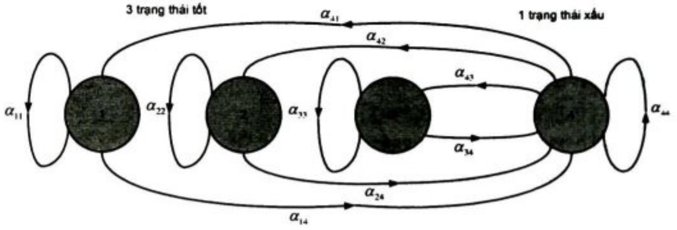
của kênh có các lỗi cụm. Các tham số của mô hình có thể được ước tính từ dữ liệu đo hoặc mô phỏng bằng các thủ tục sẽ được khảo sát trong các phần sau đây.

Mô hình Fritchman lần đầu tiên được đề xuất vào năm 1967, hiện đang được quan tâm nhiều, vì mô hình Fritchman tỏ ra phù hợp với việc mô hình hóa các *lỗi cụm* trong các kênh vô tuyến di động và cũng vì nó dễ dàng ước tính các tham số của mô hình Fritchman từ các phân bố lỗi cụm. Đối với các kênh nhị phân, cốt lõi của Fritchman là phân chia không gian trạng thái thành *k* trạng thái tốt và *N-k* trạng thái xấu. Các trạng thái tốt thể hiện các truyền dẫn không lỗi và các trạng thái xấu biểu thị lỗi truyền dẫn. Vì vậy, các thực thể trong ma trận B là 0 và 1 và không nhất thiết phải ước tính chúng. Mô hình Fritchman với ba trạng thái tốt và một trạng thái xấu được minh họa trong hình 15.9. Mô hình này có ma trận chuyển dịch trạng thái:

$$A = \begin{bmatrix} a_{11} & 0 & 0 & a_{14} \\ 0 & a_{22} & 0 & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \quad (15.47)$$

Ma trận B cho trường hợp này có dạng đơn giản là:

$$B = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (15.48)$$



Hình 15.9: Mô hình Fritchman có ba trạng thái tốt và một trạng thái xấu

Tổng quát, ma trận chuyển dịch trạng thái *A* cho mô hình Fritchman có thể được phân chia hóa như sau:

$$A = \begin{bmatrix} A_{gg} & A_{gb} \\ A_{bg} & A_{bb} \end{bmatrix} \quad (15.49)$$

Trong đó các ma trận con biểu diễn các xác suất chuyển dịch giữa các trạng thái tốt và xấu. Với mô hình này, có thể rút ra biểu thức giải tích cho các phân bố lỗi cụm theo các tham số mô hình và dùng các biểu thức này để ước tính các tham số của mô hình từ các phân bố lỗi cụm kinh nghiệm (đo đạc hoặc mô phỏng).

Đặt $\overline{O} = \{O_1, O_2, ..., O_T\}$ là một chuỗi lỗi với $O_k = 1$ biểu thị bit phát thứ *k* phải chịu một lỗi truyền dẫn, và $O_k = 0$ biểu thị cho truyền dẫn bit thứ *k* không bị lỗi. Ngoài ra, ký hiệu

$\{0^m \mid 1\}$ biểu thị cho sự kiện quan sát được m hoặc nhiều hơn m lần truyền dẫn không lỗi liên tiếp theo sau xảy ra một lỗi, và $\{1^m \mid 0\}$ biểu thị cho sự kiện quan sát m hoặc nhiều hơn m lỗi liên tiếp theo sau một lần truyền không lỗi. Fritchman chỉ ra rằng, các xác suất xảy ra hai sự kiện này được cho bởi tổng của các hàm mũ được đánh trọng số dưới đây:

$$\Pr\{0^m \mid 1\} = \sum_{i=1}^{k} f_i \lambda_i^{m-1} \tag{15.50}$$

$$\Pr\{1^m \mid 0\} = \sum_{i=k+1}^{N} f_i \lambda_i^{m-1} \tag{15.51}$$

Trong đó λ_i, $i = 1, 2, ..., k$ và λ_i, $i = k + 1, k + 2, ..., N$ lần lượt là các giá trị riêng của A_{gg} và A_{bb}, và tương ứng với giá trị của f_i là các hàm của a_{ij}. Từ (15.50) và (15.51) ta có thể có được xác suất tìm được chính xác m số 0 (hoặc số 1) như sau:

$$\Pr\left(0^{m-1} \mid 1\right) - \Pr\left(0^m \mid 1\right) = \sum_{i=1}^{k} f_i \lambda_i^{m-2} \left(1 - \lambda_i\right) \tag{15.52}$$

$$\Pr\left(1^{m-1} \mid 0\right) - \Pr\left(1^m \mid 0\right) = \sum_{i=k+1}^{N} f_i \lambda_i^{m-2} \left(1 - \lambda_i\right) \tag{15.53}$$

Mô hình Fritchman có thể được hiểu là tương đương với một quá trình Markov có một ma trận xác suất chuyển dịch trạng thái:

$$\bar{A} = \begin{bmatrix} \Lambda_{gg} & A_{gb} \\ A_{bg} & \Lambda_{gg} \end{bmatrix} \tag{15.54}$$

Trong đó Λ_{gg} *vμ* Λ_{bb} là các ma trận đường chéo được cho bởi:

$$\Lambda_{gg} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_k \end{bmatrix} \tag{15.55}$$

$$\Lambda_{bb} = \begin{bmatrix} \lambda_{k+1} & 0 & \cdots & 0 \\ 0 & \lambda_{k+2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \tag{15.56}$$

Lưu ý rằng trong mô hình tương đương này, không có các chuyển dịch trống tập của k trạng thái tốt và không có các chuyển dịch trong tập của N - k trạng thái xấu. Các chuyển dịch đó không thể phân biệt được từ chuỗi lỗi quan sát, vì quá trình chuyển dịch của một trạng thái tốt này sang một trạng thái tốt khác vẫn không tạo ra các lỗi và quá trình chuyển dịch không thể quan sát được từ đầu ra. Cấu trúc này được nhắc lại khi xét các *mô hình bán Markov* ở phần sau.

Mô hình Fritchman không duy nhất ngoại trừ khi chỉ có một trạng thái xấu. Trường hợp tổng quát, đây cũng là lý do mà phân bố liên tiếp không lỗi $\Pr\{0^m \mid 1\}$ và phân bố lỗi cụm (liên

tiếp m lỗi) $\Pr\{1^m \mid 0\}$ không định rõ tính phụ thuộc thống kê của các cụm lỗi và liên tục không lỗi. Trong trường hợp mô hình trạng thái đơn lỗi như hình 15.9, Fritchman chỉ ra rằng:

$$\Pr\left(0^m \mid 1\right) = \sum_{k=1}^{N-1} \frac{a_{Nk}.\left(a_{kk}\right)^m}{a_{kk}} \tag{15.57}$$

Từ (15.57) ta có thể thấy rằng, trường hợp mô hình trạng thái đơn lỗi, chuỗi không lỗi định rõ duy nhất 2(*N* - 1) tham số mô hình. Dùng thủ tục kinh nghiệm để làm cho phù hợp (N - 1) hỗn hợp hàm mũ như trong (15.50) và (15.51) với phân bố chuỗi không lỗi (được mô phỏng hoặc được đo đạc).

Trong khi mô hình Fritchman có thể ứng dụng cho các kênh rời rạc có các phân bố lỗi cụm đơn giản, thì nó có thể không phù hợp để đặc tính hóa các mẫu lỗi cụm rất phức tạp (cần có nhiều trạng thái lỗi trong mô hình). Trong những trường hợp này, rất khó ước tính các tham số mô hình từ chỉ riêng các phân bố lỗi cụm. May thay, có thể tìm ước tính khả năng giống nhất ML của các tham số bằng cách dùng các kỹ thuật lặp.

15.5. Ước tính các tham số của mô hình Markov

Mô hình Markov cho kênh rời rạc được mô tả bởi một ma trận chuyển dịch trạng thái *A* kích cỡ $N \times N$ và ma trận tạo xác suất lỗi *B* kích cỡ $M \times N$. Tìm được một thủ tục lặp để ước tính các tham số $\Gamma = \{A, B\}$ này từ một chuỗi lỗi cho trước thông qua việc mô phỏng hoặc đo đạc $\overline{O} = \{O_1, O_2, ..., O_T\}$ trên cơ sở giải thuật Baum-Welch. Thuật toán lặp này được thiết kế để hội tụ đến bộ ước tính khả năng giống nhất ML của $\Gamma = \{A, B\}$ mà làm cực đại hóa $\Pr(\overline{O} \mid \Gamma)$.

Mục đích là tính toán các ước tính của các phần tử ma trận chuyển dịch trạng thái A. Chúng được cho bởi:

$$\hat{a}_{ij} = \frac{\text{Số lần chuyển dịch được mong đợi từ trạng thái } i \text{ đến trạng thái } j}{\text{Số lần chuyển dịch được mong đợi từ trạng thái } i} \tag{15.58}$$

Ta cũng cần có các ước tính của các phần tử của ma trận tạo lỗi B, được cho bởi:

$$\hat{b}_j\left(e_k\right) = \frac{\text{Số lần được mong đợi } e_k \text{ được phát từ trạng thái } j}{\text{Số lần thăm lại được mong đợi đến trạng thái } j} \tag{15.59}$$

Các tính toán cần thiết để thực thi thuật toán Baum-Welch được mô tả trong các bước dưới đây. Ngoài ra, ta cũng triển khai chương trình Matlab để thực hiện thuật toán Baum-Welch. Hai phiên bản mã mã chương trình Matlab thực hiện tính toán chi tiết. *Phiên bản đầu* chỉ rõ "*mã chương trình cơ bản*" chứa nhiều vòng lặp và vì thế nó không hiệu quả. Tuy nhiên, nó dễ dàng liên hệ mã chương trình cơ bản này với các phương trình được định nghĩa. *Phiên bản sau* thể hiện "*hiệu quả hơn*", loại bỏ một vài hoặc tất cả các hoạt động vòng lặp để nhận được các ưu điểm về khả năng thực hiện các phép toán véc tơ của Matlab. Cần lưu ý rằng, trong một vài trường hợp mã chương trình Matlab không được *véc tơ hóa* hoàn toàn vì mục đích để dễ hiểu mã chương trình hơn. Phiên bản của thuật toán Baum-Welch được cho ở phụ lục 15A được triển khai bằng cách kết hợp các đoạn mã được véc tơ hóa.

Bước 0: Bắt đầu bằng một mô hình (giả định) khởi tạo $\Gamma = \{A, B\}$

Bước 1: Với $\Gamma = \{A, B\}$ như một mô hình, trước tiên ta tính toán "*các biến tiến*":

$$\alpha_t(i) = \Pr\left[O_1, O_2, ...O_t, s_t = i \middle| \Gamma\right] \tag{15.60}$$

Và "*các biến lùi*":

$$\beta_t = \Pr\left[O_{t+1}, O_{t+2}, ..., O_T \middle| s_t = i, \Gamma\right] \tag{15.61}$$

Với $t = 1, 2, ..., T$ và $i = 1, 2, ..., N$.

Chi tiết về các tính toán này như sau:

Các biến tiến: Tính toán các biến tiến bao gồm 3 bước: khởi tạo, quy nạp, kết thúc.

Khởi tạo:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad i = 1, 2, ..., N \tag{15.62}$$

Quy nạp:

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j\left(O_{t+1}\right), \quad 1 \le t \le T-1, \quad 1 \le j \le N \tag{15.63}$$

Kết thúc:

$$\Pr\left[\overline{O} \mid \Gamma\right] = \sum_{i=1}^{N} \alpha_T(i) \beta_T(i) \tag{15.64}$$

Lưu ý rằng

$$\sum_{i=1}^{N} \alpha_T(i) = \sum_{i=1}^{N} \Pr\left[O_1, ..., O_T, s_T = i \middle| \Gamma\right] = \Pr\left[\overline{O} \middle| \Gamma\right] \tag{15.65}$$

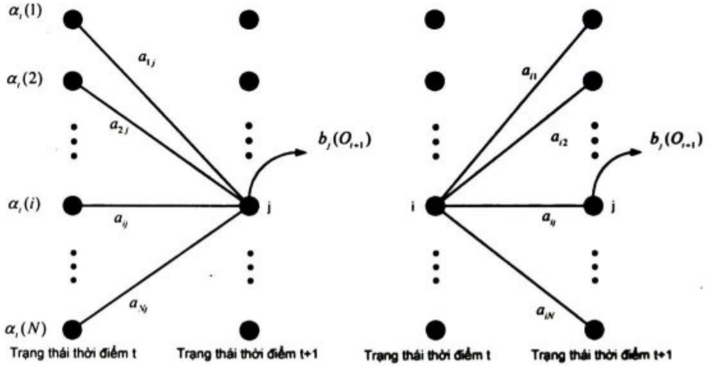
Mã chương trình Matlab cơ bản để thực hiện các tính toán này như sau: (lưu ý dùng hệ số tỉ lệ. Hệ số tỉ lệ được mô tả trong phần sau)

```
%Phiên bản 1:
alpha                   =    zeros(len,state);    % Phân bổ bộ nhớ
for column=1:states
    alpha(1,column) =    pye(1,column)*b(1,column);
    hspace*50em                                   % Khởi tạo
end

scale(1)     = sum(alpha(1,:));                   % Hệ số chuẩn hóa
alpha(1,:)   = alpha(1,:)/scale(1,:);             % chuẩn hóa
sum1=0;

for t=1:(len-1)                                   % Quy nap
    for j=1:states
        for i=1:states
            inner_sum   = alpha(t,i)*p(i,j);
            sum1        = sum1  +   inner_sum;
        end
```

```
        alpha(t+1,j)        = sum1*b(out(t+1)+1,j);
        sum1                = 0;
    end
    scale(t+1)              = sum(alpha(t+1,:));       % Hệ số chuẩn hóa
    alpha(t+1,:)            = alpha(t+1,:)/scale(t+1);% Chuẩn hóa
end
```

Mã chương trình Matlab hiệu quả hơn bằng cách khử một số vòng lặp như đã đề cập ở trên.

```
%Phiên bản 2:
alpha         = zeros(len,states);                      % Phân bổ bộ nhớ
alpha(1,:)    = pye.*b(1,:);                            % Khởi tạo
scale(1)      = sum(alpha(1,:));                        % Hệ số chuẩn hóa
alpha(1,:)    = alpha(1,:)/scale(1);                    % Chuẩn hóa

for t=1:len-1                                           % induction
    alpha(t+1,:)      = (alpha(t,:)*p).*b(out(t+1)+1,:);
    scale(t+1)        = sum(alpha(t+1,:));              % Hế số chuẩn hóa
    alpha(t+1,:)      = alpha(t+1,:)/scale(t+1);        % Chuẩn hóa
end
```



Hình 15.10: Ước tính tham số cho mô hình Markov ẩn (HMM)

Các biến lùi: Tính toán các biến lùi bao gồm 2 bước: khởi tạo và quy nạp

Khởi tạo:

$$\beta_T(i) = 1, \quad i = 1,2,...,N \tag{15.66}$$

Quy nạp:

$$\beta_t(t) = \sum_{j=1}^{N} \beta_{t+1}(j) b_j(O_{t+1}) a_{ij}; \quad 1 \le t \le T-1, \; 1 \le j \le N \tag{15.67}$$

Các chi tiết về các tính toán này được minh họa trong hình 15.10. Mã chương trình Matlab cơ bản cho đoạn này như sau:

```
%Phiên bản 1:
beta          = zeros(len,states);          % Phân bổ nhớ
beta(len,:)  = 1/scale(len);               % Khởi tao và lấy tỉ lệ
for t=(len-1):-1:1                         % induction
    for i = 1:states
        for j = 1:states
            inner_sum = p(i,j)...
                *b(out(t+1)+1,j)*beta(t+1,j);
            sum2          = sum2 + inner_sum;
        end
        beta(t,i)    = sum2;
        sum2         = 0;
    end
    beta(t,:)        = beta(t,:)/scale(t); % Lấy tỉ lệ
end
```

Phiên bản hiệu quả hơn (véc tơ hóa từng phần) như sau:

```
%Phiên bản 2:
beta          = zeros(len,states);          % Phân bổ nhớ
beta(len,:)  = 1/scale(len);               % Khởi tạo và lấy tỉ lệ

for t=(len-1):-1:1                         % Quy nạp
    beta(t,:)= (beta(t+1,:).*b(out(t+1)+1,:))*(p')/scale(t);
end
```

Bước 2: Bước tiếp theo là tính toán $\gamma_t(i)$ theo

$$\gamma_t(i) = \Pr\left[s_t = i \middle| \bar{O}, \Gamma\right] = \frac{\alpha_t(i)\beta_t(i)}{\Pr\left[\bar{O} \mid \Gamma\right]}; \quad i = 1, 2, \ldots, N \tag{15.68}$$

Mã chương trình Matlab cơ bản để thực hiện tính toán này như sau:

```
%Phiên bản 1:
gamma          = zeros(len,states);          % Phân bổ nhớ
for i=1:len
    for j=1:states
        gamma(i,j)   = alpha(i,j)*beta(i,j);% Tính toán biến gamma variable
    end
    gamma(i,:)=gamma(i,:)/sum(gamma(i,:));
end
```

Ta thấy rằng, không nhất thiết phải lưu giữ tất cả các giá trị này trong bộ nhớ, vì chúng có thể được lấy tổng đối với các chỉ số thời gian. Đoạn mã chương trình Matlab dưới đây giải thích rõ hơn về điều này.

```
%Phiên bản 2:
gamma_sum       = zeros(1,states);                  % Phân bổ nhớ
for t = 1:len
    gamma_sum   = gamma_sum + alpha(t,:).*beta(t,:);
end
```

Đại lượng $\xi_t(i,j)$ được định nghĩa bởi:

$$\xi_t(i,j) = \Pr\left[s_t = i, s_{t+1} = j \middle| \bar{O}, \Gamma\right] = \frac{\alpha_t(i).a_{ij}.b_j(O_{t+1}).\beta_{t+1}(j)}{\Pr\left[\bar{O} \mid \Gamma\right]} \quad (15.69)$$

có thể được tính toán như sau:

```
%Phiên bản 1:
prob_model_given_seq     = zeros(1,len);        % Phân bổ nhớ cho mô hình
eta                      = zeros(states,states,len);% Phân bổ nhớ cho eta
for t=1:len                                       % Bắt đầu vòng lặp
    for i =1:states
        prob_model_given_seq     = zeros(1,len);% phân bổ bộ nhớ
        temp(i)                  = alpha(t,i)*beta(t,i);
    end
    prob_model_given_seq(t)      = sum(temp);  % Xác suất của mô hình
end

for i=1:states
    for j=1:states
        for t =1:(len-1)
            eta(i,j,t)=( (alpha(t,i)*p(i,j)...
                *b(out(t+1)+1),j)*beta(t+1,j)));
        end
    end
end
```

Cần phải đặc biệt lưu ý rằng, chương trình Matlab cho thấy rằng, ta không nhất thiết phải lưu lại biến số này cho tất cả các chỉ số thời gian. Đúng hơn ta có thể tính toán:

```
%Phiên bản 2:
eta             = zeros(states,states);             % Phân bổ bộ nhớ
sum_eta         = zeros(states,states);             % Phân bổ bộ nhớ

for t = 1:(len-1)
    for i=1:states
        eta(i,:) = ((alpha(t,i)*(p(i,:)...
            *(b(out(t+1)+1),:))).*beta(t+1,:)));
    end
    sum_eta     = sum_eta + eta;   % cộng với các giá trị của eta
end
```

Tại đây, ta xác định xác suất chuyển dịch trạng thái sang-trạng thái mới $\hat{a}_{ij}$ theo:

$$\hat{a}_{ij} = \frac{\text{Số lần chuyển dịch được mong đợi từ trạng thái } i \text{ đến trạng thái } j}{\text{Số lần chuyển dịch được mong đợi từ trạng thái } i} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \quad (15.70)$$

Mã chương trình Matlab cơ bản là:

```
%Phiên bản 1:
for i=1:states
    for j =1:states
        p_estimate(i,j)         =         sum(eta(i,j,:))/(sum(gamma(:,i))-
gamma(len,i));
    end
    p_estimate(i,:)= p_estimate(i,:)/sum(p_estimate(i,:));% chuẩn hóa
end
```

Hay đoạn mã chương trình:

```
%Phiên bản 2:
for i =1:sates
    for j =1:states
        p_estimate(i,j)      = sum_eta(i,j)...
            /(gamma(i)-alpha(len,i).*beta(len,i)...
            /(sum(alpha(len,:).*beta(len,:))));
    end
    p_estimate(i,:)      = p_estimate(i,:)/sum(p_estimate(i,:));
end
```

Tiếp theo $\hat{b}_j(e_k)$ được định nghĩa bởi:

$$\hat{b}_j(e_k) = \frac{\text{Số lần mong đợi } e_k \text{ được phát đi từ trạng thái } j}{\text{Số lần thăm lại mong đợi đến trạng thái } j} = \frac{\sum_{t=1 \| O_t = e_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \quad (15.71)$$

Được tính toán. Mã chương trình Matlab cơ bản là:

```
%Phiên bản 1:
for j =1:states
    i    = find(out==0);                  % Tìm các chỉ số của các biến đúng
for k = 1:length(i)
        sum_gamma          = sum_gamma + gamma(i(k),j);
    end
    b(1,j)                 = sum_gamma/sum(gamma(:,j));
    sum_gamma              = 0;
end
```

```
for j = 1:sates
    ii = find(out==1);                % Tìm các chỉ số của các lỗi
    for k = 1:length(ii)
        sum_gamma         = sum_gamma + gamma(ii(k),j);
    end
    b(2,j)                = sum_gamma/sum(gamma(:,j));
    sum_gamma             = 0;
end
for i = 1:states
    b(:,i)                = b(:,i)/sum(b(:,i));
end
```

Dùng các tính toán hiệu quả hơn, ma trận xác suất ký hiệu đầu ra có thể được ước tính là:

```
%Phiên bản 2:
out_0       = find(out==0);  % Tìm các chỉ số của các bit đúng
out_1       = find(out==1);  % Tìm các chỉ số của các lỗi
sum_0       = zeros(1,states);
sum_1       = zeros(1,states);
gamma_sum   = sum(gamma);

for i =1:length(out_0)
    sum_0   = sum_0 +gamma(out_0(i),:);   % Cộng với các bit đúng
end
for i=1:states
    for j =1:2
        if j==1
           b(j,i)  = sum_0(i)/gamma_sum(i); % Các phần tử b,các bit lỗi
        end
    end
end
for i = 1:states
    b(:,i)  = b(:,i)/sum(b(:,i));
end
```

Ta cũng có thể tính

$$\hat{\pi} = \left(\text{số lần mong đợi ở trạng thái } S_i \text{ tại thời điểm } t = 1\right) = \alpha_1(i).\beta_1(i) \quad (15.72)$$

Bước 3: Trở lại bước 1 với các giá trị mới của $\hat{\Gamma} = \left\{\hat{A}, \hat{B}, \hat{\pi}\right\}$, hay tương đương $\hat{\Gamma} = \Gamma$, đạt được bước 2 và lặp lại đến khi đạt được mức hội tụ mong muốn như được đề cập dưới đây.

15.5.1. Lấy tỉ lệ

Các véc tơ tiến và lùi có khuynh hướng tiến tới mũ 0 khi kích thước dữ liệu lớn và phải được lấy tỉ lệ phù hợp để ngăn ngừa thiếu hụt số hạng. Hằng số tỉ lệ, cách thức dùng nó được thực hiện trong mã chương trình Matlab được cho ở file **NVD15_bwa.m** trong phụ lục 15A, trước tiên được định nghĩa là:

$$C_t = \sum_{i=1}^{N} \alpha_t(i) \tag{15.73}$$

Các giá trị của $\alpha_t(j)$ được lấy tỉ lệ, được ký hiệu là $\overline{\alpha}_t(j)$ và được cho bởi:

$$\overline{\alpha}_t(j) = \frac{\alpha_t(j)}{C_t} \tag{15.74}$$

Tất nhiên, điều này có nghĩa là:

$$\sum_{i=1}^{N} \overline{\alpha}_t(i) = 1 \tag{15.75}$$

Các giá trị của C_t được lưu lại và được dùng để lấy tỉ lệ các biến lùi. Các giá trị được lấy tỉ lệ của $\beta_t(i)$, được ký hiệu là $\overline{\beta}_t$, được cho bởi:

$$\overline{\beta}_t(i) = \frac{\beta_t(i)}{C_t} \tag{15.76}$$

Với giá trị khởi tạo:

$$\overline{\beta}_T = \frac{[\mathbf{1}]}{C_T}$$

Trong đó [1] ký hiệu cho véc tơ cột chứa tất cả số 1. Biến γ cũng có thể được chuẩn hóa nếu muốn, nhưng việc lấy tỉ lệ các biến γ là không cần thiết.

15.5.2. Tiêu chuẩn ngừng và mức độ hội tụ

Vì thuật toán Baum-Welch là lặp, nên cần phải xác định số lần lặp cần thiết để đáp ứng mức độ chính xác của mô hình. Có lẽ cách tốt nhất để thực hiện điều này là hiển thị các ước tính của A và B khi thuật toán đang thực hiện. Nếu muốn mỗi phần tử của A và B chính xác đến một số con số ý nghĩa, thì cho phép liên tục thực hiện thuật toán cho tới khi các phần tử của A và B không còn thay đổi nữa, trong mức độ chính xác cho trước, từ lần lặp này tới lần lặp khác. Sau đó, thuật toán được kết thúc bằng nhân công. Kỹ thuật này khá hấp dẫn, vì biết được mức độ chính xác. Hơn nữa, dựa vào hiểu biết có thể đơn giản hóa một số lần lặp.

Phương pháp được dùng phổ biến khác để xác định tính hội tụ là liên tục lặp cho tới khi các giá trị kế tiếp của $\Pr\left[\overline{O} \mid \Gamma\right]$ lệch nhau rất nhỏ. (Thuật toán Baum-Welch đảm bảo hội tụ tới nghiệm khả năng giống nhất ML. Chứng minh điều này được thấy ở nhiều tài liệu). Giá trị của $\Pr\left[\overline{O} \mid \Gamma\right]$ được xác định theo hằng số tỉ lệ Ct trong (15.73). Cụ thể:

$$\Pr\left[\overline{O} \mid \Gamma\right] = \prod_{t=1}^{T} C_t \tag{15.77}$$

Khi T lớn, con số này sẽ rất nhỏ và thường được biểu diễn là:

$$\log_{10} \Pr\left[\overline{O} \mid \Gamma\right] = \sum_{t=1}^{T} \log_{10} C_t \tag{15.78}$$

Được xem là tỉ lệ log-ML (lấy log của hàm khả năng giống nhất) và được minh họa ở trường hợp 1 trong phần 15.6.

Cũng cần chỉ ra rằng, đối với tập dữ liệu cho trước, các ước tính của A và B là không duy nhất trừ khi đã định rõ các ước tính khởi tạo của A, B và Π. Vì véc tơ lỗi là một hàm của A và B, nên các kết hợp khác nhau có thể tạo ra các kết quả thống kê tương đương và kết quả cụ thể sẽ phụ thuộc vào các điều kiện khởi tạo (điều kiện đầu).

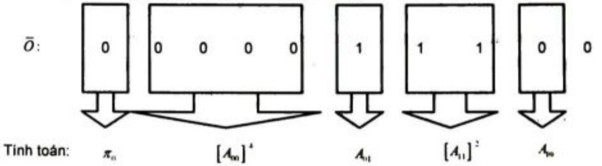
15.5.3. Mô hình Markov tương đương khối

Thuật toán Baum-Welch là một trong nhiều thuật toán *ước tính lại* khả dụng để tính toán các tham số của mô hình dựa trên véc tơ lỗi. Chuỗi lỗi này có thể kéo theo hàng nghìn, hoặc thậm chí hàng triệu ký hiệu. Lưu ý rằng, để ước tính chính xác giá trị nhỏ của a_{ij} thì chu kỳ quan sát phải đủ lâu. Trong các hệ thống truyền thông, các giá trị nhỏ này thường là giá trị tới hạn để ước tính hiệu năng hệ thống. Độ phức tạp tính toán liên quan đến thuật toán Baum - Welch khá lớn khi véc tơ lỗi dài, vì các biến tiến và lùi được tính toán đối với mỗi ký hiệu trong chuỗi lỗi đã cho. Hơn nữa, tính hội tụ chậm làm tăng thêm độ phức tạp tính toán. Với các ứng dụng xác suất lỗi thấp, véc tơ lỗi có chuỗi dài liên tiếp số 0. Ngoài ra, véc tơ lỗi phải chứa một số lượng đáng kể các sự kiện lỗi để véc tơ lỗi định nghĩa chính xác mô hình. Trong các tình huống này, thuật toán Baum-Welch là không hiệu quả và cần phải có một thuật toán nhanh hơn.

Phương pháp Turin: Một phương pháp được Turin đề xuất để giải quyết vấn đề này, bao gồm quá trình tính toán các biến tiến và biến lùi dùng phiên bản ma trận khối dạng:

$$A = \begin{bmatrix} A_{00} & A_{01} \\ A_{10} & A_{11} \end{bmatrix} \tag{15.79}$$

Quan hệ giữa các số hạng trong ma trận được định nghĩa bởi (15.79) được minh họa ở hình 15.11. Lưu ý rằng, số mũ của các ma trận con trong các tính toán có thể được tính toán trước (tiền tính toán-precompute) và được dùng *lại* để giảm toàn bộ tải tính toán.



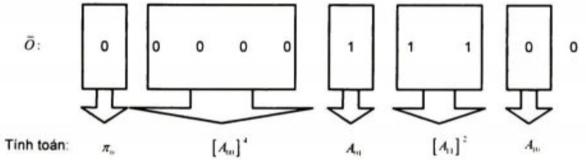
Hình 15.11: Các tính toán cho chuỗi 0000011100 (phiên bản 1)

Phương pháp Sivaprakasam và Shanmugan: Một thay đổi khác được Sivaprakasam và Shanmugan đề xuất dựa trên thực tế rằng đối với mô hình Markov tổng quát là có mô hình Fritcham thống kê tương đương có *k* trạng thái tốt và *N-k* trạng thái xấu và một ma trận A có dạng:

$$A = \begin{bmatrix} \Lambda_{00} & A_{01} \\ A_{10} & \Lambda_{11} \end{bmatrix} \tag{15.80}$$

Trong đó Λ_{00} và Λ_{11} là các ma trận đường chéo. Các mô hình này thường được coi là các mô hình Markov đường chéo khối. Với mô hình này, kênh giữ nguyên cùng trạng thái trong

khoảng thời gian một cụm lỗi và chỉ thay đổi trạng thái ở thời điểm cuối một cụm. Kết quả là, tất cả các biến chỉ được tính toán tại các bước thời gian làm thay đổi ký hiệu lỗi. Nói cách khác, các biến được tính toán tại thời điểm bắt đầu của mỗi cụm các lỗi chứ không phải là trên mỗi ký hiệu. Bằng cách thay đổi này, các tính toán có dạng được minh họa ở hình 15.12. Tại đây, lưu ý rằng các tính toán trong khoảng thời gian các cụm dài liên tiếp số 1 và số 0 liên quan với việc làm tăng lũy thừa (số mũ) của ma trận đường chéo chứ không phải là làm tăng lũy thừa của các ma trận bất kỳ, và các phép nhân ma trận chỉ xảy ra tại thời điểm chuyển từ cụm này sang cụm khác. Vì vậy, rất hiệu quả tính toán, và các cụm lỗi dài có thể được xử lý mà không phải tính toán và lưu trữ quá mức.



Hình 15.12: Các tính toán cho chuỗi 0000011100 (phiên bản 2)

Sử dụng mô hình này ta chỉ cần quan tâm các độ dài liên tiếp các số 1 hoặc 0. Vì vậy, véc tơ lỗi có thể được đặt trong dạng khối rất kết đặc. Như một ví dụ minh họa đơn giản là, xét một chuỗi lỗi:

$$E = \left[\underbrace{00000}_{0^5}\underbrace{111}_{1^3}\underbrace{00000000000000}_{0^{14}}\underbrace{11}_{1^2}\underbrace{000000000000000000000}_{0^{21}}1\underbrace{00}_{0^2}\right] \tag{15.81}$$

Véc tơ độ dài liên tiếp tương ứng chuỗi lỗi này được viết là:

$$V = \left[0^5 1^3 0^{14} 1^2 0^{21} 1 0^2\right] \tag{15.82}$$

Mã chương trình Matlab **NVD15_seglength.m** được cho ở phụ lục 15A, tạo ra véc tơ độ dài liên tiếp được định nghĩa bởi (15.82) từ véc tơ lỗi được định nghĩa bởi (15.81).

Việc triển khai các giải thuật ước tính ***lại*** dựa trên các mô hình Markov tương đương khối không được đề cập ở đây. Chương trình Matlab dựa trên công trình của Sivaprakasam và Shanmugan được cho ở file **NVD15_SemiMarkov.m** trong phụ lục 15A. Ví dụ minh họa cách dùng thuật toán này được trình bày ở phần sau

Không kể đến các thuật toán đã được dùng, mô hình Markov cho kênh rời rạc phải được tính toán cho các giá trị khác nhau của các tham số của kênh vật lý cơ bản. *Nếu kênh vật lý lớp dưới bị thay đổi theo bất kỳ cách nào, thì mô hình kênh rời rạc phải được ước tính lại.* Ví dụ, nếu một mô hình kênh rời rạc được triển khai cho một kênh đã cho, thì mô hình Markov phải được triển khai từ các mô phỏng mức dạng sóng hoặc các đo đạc cho các giá trị khác nhau của các tham số kênh bao gồm cả SNR. Vì vậy, tập các mô hình Markov được thông số hóa cho kênh lớp dưới có thể được triển khai và được sử dụng để thiết kế và phân tích các bộ mã hóa kiểm soát lỗi, các bộ đan xen, v.v...

15.6. Hai trường hợp điển hình

Ta kết thúc chương bằng hai trường hợp điển hình với mục đích làm sáng tỏ việc xác định kênh Markov và xác định mô hình bán Markov như sau:

Trường hợp 1, véc tơ lỗi được tạo ra trong ví dụ 15.3 được dùng làm đầu vào thuật toán Baum-Welch. Kết quả là một mô hình kênh được ước tính. Kết quả phân bố độ dài liên tục và xác suất lỗi từ mô hình này được xác định và sau đó được so sánh với phân bố độ dài liên tục và xác suất lỗi của chuỗi gốc được tạo ra.

Trường hợp 2, về cơ bản giống như trường hợp 1. Khác nhau chủ yếu là dùng mô hình *bán* Markov tương đương khối chứ không phải là mô hình Markov. Động cơ để dùng mô hình bán Markov là, mô hình bán Markov làm giảm đáng kể thời gian cần thiết để rút ra mô hình từ dữ liệu mô phỏng hoặc dữ liệu đo.

15.6.1. Trường hợp 1:

Trong trường hợp này, ta tạo ra một chuỗi lỗi có chiều dài L = 20.000 với một mô hình 2 trạng thái cho trước được định nghĩa bởi ma trận A và B sau:

$$A = \begin{bmatrix} 0{,}95 & 0{,}05 \\ 0{,}10 & 0{,}90 \end{bmatrix} \tag{15.83}$$

$$B = \begin{bmatrix} 0{,}95 & 0{,}50 \\ 0{,}05 & 0{,}50 \end{bmatrix} \tag{15.84}$$

Sau đó thuật toán Baum-Welch được dùng để ước tính mô hình sử dụng chuỗi lỗi đã được tạo ra. Lưu ý rằng ta, ta thừa nhận một mô hình ba trạng thái khi ứng dụng mô hình Baum-Welch. Điều này có thể dường như lạ thường nhưng, cho trước dữ liệu đo hoặc mô phỏng, thì mô hình kênh tạo ra dữ liệu không được biết. Hội thoại Matlab, với một số hội thoại không cần thiết được loại bỏ như sau (nghiên cứu kỹ chương trình Matlab):

```
>> NVD15_errvector
    Chấp nhận các giá trị mặc định ?
    Nhập y nếu đồng ý hoặc n nếu không:              n
    Nhập số các điểm được tạo ra L:                  20000
    Nhập ma trận chuyển dịch trạng thái A:           [0,95 0,05; 0,1 0,9]
    Nhập ma trận phân bố lỗi B:                      [0,95 0,5; 0,05 0,5]
>> Đầu ra 1 = đầu ra;
>> NVD15_bwa(20,3,đầu ra)
Nhập ma trận chuyển dịch trạng thái khởi tạo P:
    [0,9 0,05 0,05; 0,1 0,8 0,1; 0,1 0,2 0,7]
Nhập véc tơ xác suất trạng thái khởi tạo pye: [0,3 0,3 0,4]
Nhập ma trận xác suất ký hiệu đầu ra khởi tạo B: [0,9 0,8 0,7; 0,1 0,2 0,3]
```

Sau một lần lặp ta có (cần lưu ý rằng, chạy lại chương trình này có thể dẫn đến các ước tính cho A và B có thể khác với kết quả ở đây, thậm chí sử dụng cùng tham số các điều kiện

đầu. Lý do là véc tơ lỗi dựa vào các ước tính của A và B là một hàm mẫu của quá trình ngẫu nhiên. Ngoài ra, các hàng của $\hat{A}$ và các cột của $\hat{B}$ có thể có tổng không phải là 1 khi mà các phần tử của $\hat{A}$ và $\hat{B}$ được biểu diễn với độ chính xác hạn chế).

$$\hat{A}_1 = \begin{bmatrix} 0,9051 & 0,0469 & 0,0481 \\ 0,0991 & 0,7931 & 0,1078 \\ 0,0895 & 0,1856 & 0,7249 \end{bmatrix} \qquad \hat{B}_1 = \begin{bmatrix} 0,9206 & 0,7462 & 0,5848 \\ 0,0794 & 0,2538 & 0,4152 \end{bmatrix}$$

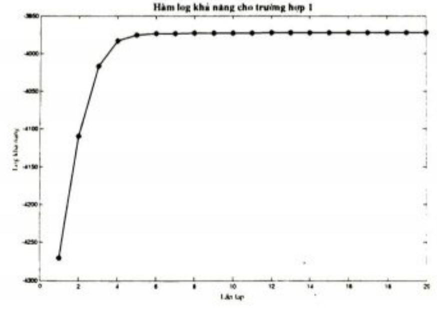
Sau 10 lần lặp ma trận chuyển dịch trạng thái được ước tính là:

$$\hat{A}_{10} = \begin{bmatrix} 0,9415 & 0,0300 & 0,0285 \\ 0,1157 & 0,7504 & 0,1340 \\ 0,0669 & 0,1382 & 0,7949 \end{bmatrix} \qquad \hat{B}_{10} = \begin{bmatrix} 0,9547 & 0,6745 & 0,4157 \\ 0,0453 & 0,3255 & 0,5743 \end{bmatrix}$$

Tại lần lặp thứ 20 là điểm kết thúc, ma trận chuyển dịch trạng thái được ước tính là:

$$\hat{A}_{20} = \begin{bmatrix} 0,9437 & 0,0299 & 0,0263 \\ 0,1173 & 0,7425 & 0,1402 \\ 0,0663 & 0,1324 & 0,8013 \end{bmatrix} \qquad \hat{B}_{20} = \begin{bmatrix} 0,9522 & 0,6637 & 0,4313 \\ 0,0478 & 0,3363 & 0,5687 \end{bmatrix}$$

Hàm log khả năng giống được minh họa ở hình 15.13. Cho thấy, hàm log khả năng hội tụ sau khoảng 10 lần lặp. Lưu ý rằng kết luận này phù hợp với các tính toán trước đây cho A_k và B_k khi $k = 10$ và $k = 20$. Cũng cần lưu ý rằng, như thảo luận trước đây, các con số khả năng giống là rất nhỏ.



Hình 15.13: Hàm log khả năng cho trường hợp 1

Tại đây, ta tạo chuỗi thứ hai *out2* dùng mô hình được ước tính bởi thuật toán Baum-Welch. Điều này cho ta với hội thoại khảo sát các giá trị mặc định được ẩn đi:

```
>> a = [0,9437 0,0299 0,0263; 0,1173 0,7425 0,1402; 0,0663 0,1324 0,8013]
>> b = [0,9522 0,6637 0,4313; 0,0478 0,3363 0,5687];
```

>> NVD15_errvector

Chấp nhận các giá trị mặc định ?

Nhập y cho yes hay n cho no > n

Nhập số các điểm được tạo ra L > 20000

Nhập ma trận chuyển dịch trạng thái A > a

Nhập ma trận phân bố lỗi B > b

>> Đầu ra 2 = đầu ra;

Để tính toán và vẽ $\Pr\left\{0^m|1\right\}$ cho cả out1 và out2, ta thực hiện chạy đoạn chương trình Matlab sau:

>> runcode1 = NVD15_seglength(out1);

>> runcode2 = NVD15_seglength(out2);

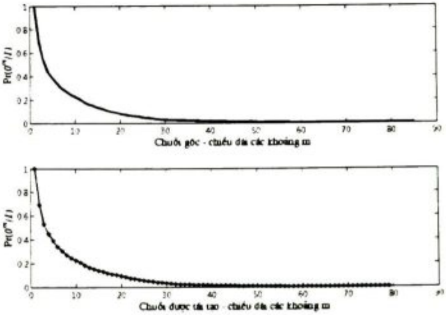
>> NVD15_interval (runcode1, runcode2);

Kết quả được minh họa ở hình 15.14. Trong đó nửa trên minh họa $\Pr\left\{0^m|1\right\}$ cho dữ liệu gốc và hình dưới minh họa $\Pr\left\{0^m|1\right\}$ dùng dữ liệu được tạo ra bởi mô hình được ước tính bởi giải thuật Baum-Welch.

Có một vài điểm khác nhau giữa hai hình được minh họa ở hình 15.14, đặc biệt trong lân cận m = 10. Tuy nhiên lưu ý rằng, xác suất lỗi được cho bởi.

>>p1 = sum(out1)/L kết quả là P1 = 0,0038

>>p2 = sum(out2)/L kết quả là P2 = 0,0035



Hình 15.14: $\Pr\left\{0^m|1\right\}$ *đối với dữ liệu gốc và dữ liệu từ mô hình được ước tính*

15.6.2. Trường hợp 2

Xét trường hợp kiểm tra được Sivaprakasam & Shammugan dùng để minh họa mô hình Markov đường chéo khối. Giả sử một mô hình ba trạng thái có hai trạng thái tối và một trạng thái xấu được định nghĩa bởi (lưu ý rằng ma trận chuyển dịch trạng thái A biểu diễn kênh vật lý

và không có dạng đường chéo khối. Ma trận chuyển dịch trạng thái được ước tính của A là $\hat{A}$ sẽ có dạng đường chéo khối):

$$A = \begin{bmatrix} 0,90 & 0,02 & 0,08 \\ 0,10 & 0,50 & 0,40 \\ 0,10 & 0,20 & 0,70 \end{bmatrix} \tag{15.85}$$

Vì các trạng thái tốt là không lỗi và các lỗi luôn được tạo ra trong trạng thái xấu, nên ma trận quan sát lỗi được định nghĩa bởi:

$$B = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{15.86}$$

Lưu ý rằng, mô hình này là mô hình *bán ẩn*. Nếu xảy ra một lỗi, ta biết rằng lỗi đó được tạo ra bởi trạng thái 3. Tuy nhiên nếu lỗi không được tạo ra, thì không nhận biết được trạng thái đó.

Trước hết, ta tạo một véc tơ lỗi có độ dài 200.000, với ma trận chuyển dịch trạng thái A và ma trận quan sát lỗi B. Các câu lệnh Matlab, với ẩn hội thoại không cần thiết như sau.

```
>> a1 = [0,90 0,02 0,08; 0,10 0,50 0,40; 0,10 0,20 0,70];
>> b1 = [1 1 1; 0 0 1];
>> NVD15_errvector
Chấp nhận các giá trị mặc định?
    Nhập y cho yes hay n cho no > n
    Nhập số các điểm được tạo ra L > 20000
    Nhập ma trận chuyển dịch trạng thái A > a1
    Nhập ma trận phân bố lỗi B > b1
>> out1 = out;
>> runcode1 = NVD15_seglenght(out1)
>>[A_matrix, pi_est] = NVD15_semiMarkov(runcode1,100,[2 1]);
```

Lưu ý rằng, ta phải chạy chương trình **NVD15_seglenght** trước khi chạy chương trình **NVD15_semiMarkov** để tạo véc tơ độ dài liên tục.

Sau 50 lần lặp, ước tính ma trận chuyển dịch trạng thái là:

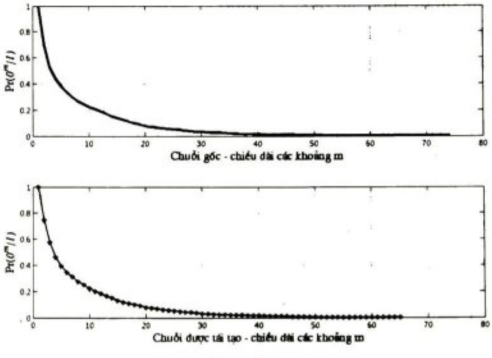
$$\hat{A} = \begin{bmatrix} 0,9043 & 0,0 & 0,0957 \\ 0,0 & 0,4911 & 0,5089 \\ 0,1402 & 0,1515 & 0,7083 \end{bmatrix} \tag{15.87}$$

Lưu ý rằng đây là gần sát với kết quả đạt được bởi Sivaprakasam & Shammugan . Có một số giải thích cho các khác nhau này. *Trước hết*, véc tơ lỗi được dựa trên mô hình được ước tính là một hàm mẫu của quá trình stochastic và với mục đích thực tế thì hai véc tơ lỗi là không đồng nhất. *Ngoài ra*, các véc tơ lỗi có độ dài khác nhau sẽ dẫn đến các môi hình hơi khác nhau. *Cuối cùng*, ta đã cố định số lần lặp và Sivaprakasam & Shammugan dùng tiêu chuẩn ngừng khác.

Bước tiếp theo là tạo véc tơ lỗi thứ hai dựa trên mô hình được ước tính và vẽ $\Pr\{0^m|1\}$ cho cả véc tơ lỗi gốc và véc tơ lỗi được tạo ra bởi mô hình được ước tính. Điều này được thực hiện dùng mã chương trình Matlab dưới đây.

```
>> a2 = [0.9043 0.0 0.0957 0.04911 0.45089; 0.1402 0.1515 0.7083];
>> b2 = [1 1 0;0 0 1];
>> NVD15_errvector
      Chấp nhận các giá trị mặc định ?
      Nhập y cho yes hay n cho no > n
      Nhập số các điểm được tạo ra L > 20000
      Nhập ma trận chuyển dịch trạng thái A > a2
      Nhập ma trận phân bố lỗi B > b2
>> out2 = out;
>> runcode2 = NVD15_seglenght(out2)
>> NVD15_intervals2 (runcode1, runcode2);
```

Các kết quả được minh họa ở hình 15.15. Lưu ý rằng, các kết quả là gần giống dù là mô hình gốc được định nghĩa bởi A và mô hình được ước tính được định nghĩa bởi $\hat{A}$ là hơi khác. Rõ rằng, ta biết từ bắt đầu bài toán này là A và $\hat{A}$ có thể sẽ khác vì $\hat{A}$ được giam giữ có dạng đường chéo khối.



Hình 15.15: $\Pr\{0^m|1\}$ *cho trường hợp 2*

Cũng là bài học để kiểm tra các xác suất lỗi. Để thực hiện điều này ta tính toán

>> pe1 = sum(out1)/20000; kết quả là Pe1 = 0.3617;

>> pe2 = sum(out2)/20000; kết quả là Pe2 = 0.3538;

Một khi thực hiện lại ta có giá trị gần giống.

Chương 16

KỸ THUẬT MÔ PHỎNG HIỆU QUẢ

16.1. Mở đầu

Kỹ thuật mô phỏng Monte Carlo cơ bản đã được nghiên cứu và ứng dụng vào một số hệ thống truyền thông đơn giản trong chương 9 và chương 10, ở đó đã cho thấy phương pháp Monte Carlo có khả năng ứng dụng cho tất cả các hệ thống bất kể tính phức tạp và kiến trúc của hệ thống, nhưng để đạt được ước tính hiệu năng tin cậy thì thời gian mô phỏng rất lâu, vì vậy đôi khi trở nên không thực tế.

Kỹ thuật bán phân tích SA trong chương 10 cho ta thực thi các mô phỏng rất nhanh nhưng phải biết hàm mật độ xác suất pdf.

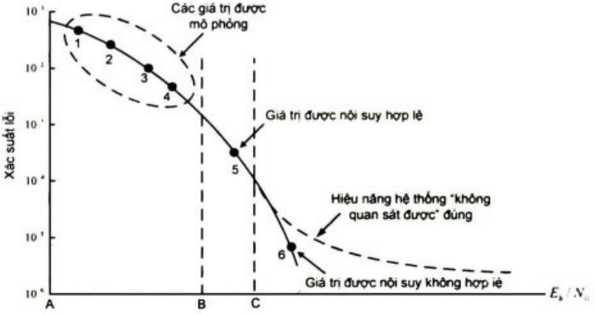
Chương này sẽ trình bày ngắn gọn các kỹ thuật mô phỏng nhằm khắc phục nhược điểm gian mô phỏng của phương pháp Monte Carlo. Ta xét ba phương pháp: ***(i)*** *Phương pháp 1,* ngoại suy đuôi là phương pháp làm cho đường cong phù hợp với các kết quả mô phỏng MC; ***(ii)*** *Phương pháp thứ 2,* là phương pháp dựa vào việc ước tính pdf của số đo quyết định thông qua ứng dụng các phương pháp moment; ***(iii)*** *Phương pháp thứ 3,* là đánh địa chỉ, lấy mẫu quan trọng, bao gồm việc làm chệch tạp âm kênh theo cách ép các lỗi quyết định hơn nữa. Lấy mẫu quan trọng là kỹ thuật làm giảm phương sai để tạo ra ước tính BER có phương sai nhỏ hơn ước tính bởi mô phỏng MC với thời gian mô phỏng tương đương. Ứng dụng các kỹ thuật giảm phương sai về cơ bản bao gồm việc dung hòa giữa thời gian chạy máy tính và phân tích. Một số kỹ thuật này được nghiên cứu và khám phá cho các kỹ thuật mô phỏng hiệu quả vẫn còn là lĩnh vực nghiên cứu hấp dẫn.

Trong 03 kỹ thuật được trình bày trong chương này, thì kỹ thuật lấy mẫu quan trọng nhận được sự quan tâm nhất trong hầu hết các ứng dụng, do đó được xét chi tiết hơn. Để ứng dụng hiệu quả các phương pháp này (đặc biệt là lấy mẫu quan trọng) cần phải có kỹ năng phân tích. Chương này sẽ chỉ xét tóm tắt của các chủ đề này.

16.2. Ngoại suy đuôi

Phương pháp này được ứng dụng bằng cách thực hiện một số mô phỏng MC sử dụng các giá trị của E_b/N_0 để được các kết quả mô phỏng chính xác với các thời gian mô phỏng hợp lý. Sau đó các kết quả này được ngoại suy tới các giá trị E_b/N_0 mà không thực tế đối với các mô phỏng MC. Tất nhiên, phải rất cẩn thận để tránh ngoại suy rơi vào vùng E_b/N_0 mà ở đó ngoại suy bị sai. Ví dụ hệ thống minh họa ở hình 16.1. Khi $A < E_b/N_0 < B$, có thể tính toán giá trị xác xuất lỗi khả tin với thời gian mô phỏng hợp lý bằng mô phỏng MC, tạo ra các điểm 1, 2, 3, 4. Với hệ thống quan tâm, ngoại suy hợp lệ đối với $B<E_b/N_0<C$, và do đó điểm 5 được ngoại suy hợp lệ. Tiếp tục ngoại suy đến điểm 6. Tuy nhiên, hệ thống thể hiện một sàn xác xuất lỗi, và kết quả là việc ngoại suy các giá trị mô phỏng trong vùng $E_b/N_0>C$ là không hợp lệ. Vì vậy, điểm 6 là không hợp lệ. Ta phải rất cẩn thận với các kết quả ngoại suy trong các vùng mà các kết quả mô phỏng hoặc thực nghiệm không khả dụng.

Giả sử ta biết (hoặc sẽ cho rằng), hàm mật độ xác suất pdf của số đo quyết định là biến ngẫu nhiên Gausơ. Khi này, xác xuất lỗi hệ thống được cho bởi $Q\left(\sqrt{g\left(E_b/N_0\right)}\right)$, trong đó $Q(x)$ là hàm Q Gausơ và $g\left(E_b/N_0\right)$ là hàm của E_b/N_o được xác định bởi các tham số hệ thống như dạng điều chế. Ta thấy rằng việc làm phù hợp các điểm mô phỏng với hàm Gausơ Q (như các điểm 1, 2, 3 và 4 trong hình 16.1) có thể cho phép ngoại suy hiệu năng chính xác đến các giá trị E_b/N_0 cao hơn.



Hình 16.1: Dùng giá trị hợp lệ và không hợp lệ của ngoại suy đuôi

Ví dụ tổng quát hơn, giả sử rằng số đo quyết định không hoàn toàn là Gausơ nhưng có thể được xấp xỉ bởi pdf của lớp mũ tổng quát.

Lớp các pdf này được định nghĩa bởi

$$f_X\left(x|v,\sigma,m\right)=\frac{v}{\sigma\,\Gamma\left(1/v\right)\,\sqrt{8}}e^{-\left|\frac{x-m}{\sigma\sqrt{2}}\right|^{v}} \tag{16.1}$$

Trong đó Γ(.) là hàm gama, m là giá trị trung bình của X, v là chế độ, σ là tham số liên quan tới việc trải rộng pdf xung quanh giá trị trung bình. Tất nhiên, khi $v = 2$ thì tham số σ là độ chệch chuẩn của X (vì các kết quả pdf phân bố Gausơ có $v = 2$). Các kết quả pdf phân bố Laplace có $v = 1$. Nếu coi rằng, giá trị trung bình bằng 0, thì có thể chỉ ra rằng tại các giá trị T (ngưỡng lỗi) lớn:

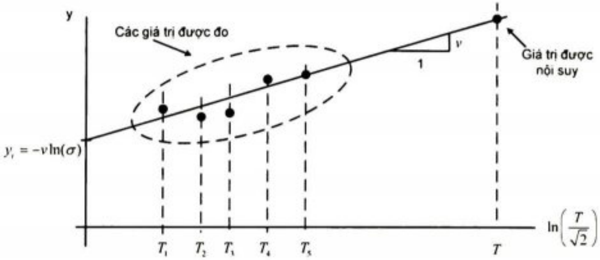
$$\int_T^{\infty} f_X\left(x|v,\sigma\right)dx \approx \mathrm{e}^{-\left(\frac{T}{\sigma\sqrt{2}}\right)^{v}} \tag{16.2}$$

Vế trái của (16.2) là BER của hệ thống khi ta dùng ngưỡng T và, vì vậy ta viết vế trái là $\hat{P}_e(T)$. Lấy logarit biểu thức (16.2) hai lần ta được:

$$\ln\left[-\ln\left(\hat{P}_e(T)\right)\right]\approx v\ln\left(\frac{T}{\sigma\sqrt{2}}\right) \tag{16.3}$$

$$\ln\left[-\ln\left(\hat{P}_e(T)\right)\right] \approx \nu\ln\left(\frac{T}{\sqrt{2}}\right) - \nu\ln(\sigma) \tag{16.4}$$

Tại đây, giả sử ta chạy *M* lần mô phỏng MC với các ngưỡng khác nhau $T_1, T_2, \ldots, T_M$, thì ước tính BER tương ứng với mỗi ngưỡng này là $\hat{P}_e(T_1), \hat{P}_e(T_2), \ldots, \hat{P}_e(T_M)$. Đồ thị của $\ln\left[-\ln\left(\hat{P}_e(T)\right)\right]$ là một hàm của $\ln(T/\sqrt{2})$ sẽ là một đường thẳng, được xác định bằng cách sử dụng hồi quy tuyến tính, với đường thẳng chắn y là $-\nu\ln(\sigma)$ và độ dốc ν. Điều này được minh hoạ trong hình 16.2.



Hình 16.2: Quá trình ngoại suy đuôi

Chế độ ν được xác định từ độ dốc. Một khi đã biết ν, thì σ được xác định từ đường thẳng chắn y là y_i theo:

$$\sigma = e^{-y_i/\sigma} \tag{16.5}$$

Một khi ν và σ được xác định thì, (16.2) được sử dụng làm bộ ước tính BER hệ thống. Ngoại suy đuôi có thể được mở rộng cho các pdf khác. Kỹ thuật này sẽ chính xác nếu có kỹ sảo phân tích, đủ để chọn một họ pdf gần đúng cho hệ thống đã cho.

16.3. Bộ ước tính pdf

Ta biết rằng xác xuất lỗi của một hệ thống truyền tin số được viết dưới dạng sau:

$$P_E = \int_T^{\infty} f_V(v)dv \tag{16.6}$$

Trong đó *T* là ngưỡng lỗi (các giá trị của *V* >*T* gây ra các lỗi) và $f_V(v)$ là pdf của con số quyết định. Việc đầu tiên để ước tính BER là ước tính pdf sau đó ước lượng tích phân được xác định trong (16.6) bằng tích phân số. Một cách trực giác là tạo một hoành đồ (tổ chức đồ dữ liệu) biểu diễn các giá trị của *V*. Để được hữu hiệu, hoành đồ này phải có đủ số bin khi *V*>*T* để cho phép thực hiện tích phân (16.6) với độ chính xác cần thiết. Hơn nữa, như ta thấy trong chương 8, hoành đồ là một bộ ước tính *chệch* khi kích thước dữ liệu *N* hữu hạn. May thay, tồn tại các lựa chọn cho phương pháp hoành đồ. Hai kỹ thuật phổ biến nhất để ước tính pdf từ một tập dữ liệu *x[n]* là chuỗi Gram-Charlier và chuỗi Parzen.

16.3.1. Phương pháp chuỗi Gram-Charlier

Chuỗi Gram-Charlier gần đúng với pdf dạng:

$$\hat{f}_Y(y) = \frac{1}{\sqrt{2\pi\sigma_v}}.e^{-\frac{y^2}{2}}.\sum_{k=1}^{N} C_k H_k(y) \tag{16.7}$$

Trong đó $H_k(.)$ biểu diễn các đa thức Chebyshev-Hermite và C_k là các hệ số chuỗi. Trong quá trình (16.7), để đơn giản ta coi rằng X là một biến ngẫu phương sai đơn vị trung bình không. Các đa thức Chebyshev-Hermite được định nghĩa bởi quan hệ hồi quy:

$$H_k(y) = yH_{k-1}(y) - (k-1)H_{k-2}(y), \quad k \geq 2 \tag{16.8}$$

Trong đó $H_0(y) = 1$ và $H_1(y) = y$. Khi cho trước các đa thức Chebyshev-Hermite, thì các hệ số có thể được tính từ:

$$C_k = \frac{1}{k!}\int_{-\infty}^{\infty} H_k(y) f_Y(y)\,dy \tag{16.9}$$

Lưu ý rằng, mặc dù $f_Y(y)$ chưa biết, nhưng (16.9) có thể được ước lượng theo các mô men của Y. Nếu Y không phải là biến ngẫu phương sai đơn vị trung bình không, thì ta có thể áp dụng chuyển đổi:

$$Z = \frac{Y - \mu_y}{\sigma_y} \tag{16.10}$$

Trong đó μ_y và σ_y biểu diễn giá trị trung bình và độ lệch chuẩn của Y. Theo biến đổi này Z trở thành biến ngẫu phương sai đơn vị trung bình không sao cho (16.7) có thể được áp dụng trực tiếp với Y được thay bởi Z.

Các chuỗi Gram-Charlier tạo ra phép gần đúng tốt cho pdf đích ở lân cận giá trị trung bình nhưng kém chính xác ở đuôi của pdf. Đáng tiếc, khi ước tính BER, BER lại là đuôi của pdf. Chuỗi Gam-Charlier có một số khó khăn khác. Một cách xấp xỉ hóa Gram-Charlier không thuộc loại không chệch tiệm cận và không hội tụ đều tới pdf khi thêm nhiều số hạng. Ngoài ra, với một số hữu hạn các số hạng, N, phép gần đúng được định nghĩa bởi (16.7) không phải là pdf đúng và thậm chí là âm đối với các giá trị đặc biệt của y. Tuy tồn tại những khiếm khuyết song chuỗi Gram-Charlier vẫn là một bộ ước tính pdf hữu hiệu cho nhiều ứng dụng.

Bộ ước tính cho các mô men của biến ngẫu nhiên tạo ra một chuỗi số liệu $x[n]$ được cho bởi:

$$E\{X^k\} = \lim_{N\to\infty}\left[\frac{1}{N}\sum_{n=1}^{N} x^k[n]\right] \tag{16.11}$$

Tất nhiên trong thực tế, kích thước số liệu N phải hữu hạn. Đáng tiếc, khi N hữu hạn thì bộ ước tính này không có các thuộc tính tối ưu. Tuy nhiên, nó là một bộ ước tính kiên định vì vậy hiệu năng thỏa đáng có thể được mong đợi khi N đủ lớn.

16.3.2. Phương pháp chuỗi Parzen:

Những khiếm khuyết của chuỗi Gram-Charlier ít nhất có thể khắc phục một phần qua việc sử dụng bộ ước tính pdf Parzen. Bộ ước tính Parzen có dạng:

$$\hat{f}_V(v) = \frac{1}{N\,h(N)}\sum_{k=1}^{N} g\left(\frac{v - v_k}{h(N)}\right) \tag{16.12}$$

Các lựa chọn *g(x)* và *h(N)* có phần tuỳ ý nhưng lựa chọn hợp lý trong nhiều ứng dụng là:

$$g(x) = \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}} \tag{16.13}$$

$$h(N) = \frac{1}{\sqrt{N}} \tag{16.14}$$

Trong (16.12), *N* là kích thước của tập số liệu, hàm *g(.)* là hàm trọng số, *h(N)* là hệ số làm nhẵn (san bằng). Mặc dù bộ ước tính Parzen bị chệch khi N hữu hạn, nhưng có thể thấy rằng nếu $h(N) \to 0$ khi $N \to \infty$ và $Nh(N) \to \infty$ khi $N \to \infty$, thì bộ ước tính Parzen là không chệch một cách tiệm cận. Rõ ràng, hàm làm nhẵn được định nghĩa bởi (16.14) thỏa mãn các yêu cầu này. Cũng có thể thấy rằng bộ ước tính Parzen là kiên định.

Cả hai kỹ thuật Gram-Charlier và Parzen đều được ứng dụng thành công cho một số bài toán. Gần đây, chúng được dùng để ước tính BER của hệ thống thông tin số hoạt động trong rất nhiều kịch bản kênh như: kênh AWGN, kênh AWGN có nhiễu đồng kênh và kênh AWGN cộng với đa đường.

16.4. Lấy mẫu quan trọng

Ta đã thấy ở chương 9 và 10, các mô phỏng MC thực hiện phép thử ngẫu nhiên một số lượng lớn các lần *N*, và đếm số các kết cục N_A, biểu thị một số sự kiện quan tâm A. Ta có thể ước tính xác xuất của sự kiện A bằng phương trình:

$$\Pr(A) = \lim_{N\to\infty}\frac{N_A}{N} \tag{16.15}$$

Giả sử N_A là số lỗi trong truyền dẫn *N* ký hiệu, thì các hệ thống truyền thông hiệu năng cao tạo ra các giá trị của N_A nhỏ hơn nhiều *N*. Đề cập ngắn gọn về các khoảng tin cậy trong chương 9 cho thấy rằng để tạo một bộ ước tính khả tin, ta cần có N_A ít nhất phải bằng 10 và tốt nhất là 100. Trong nhiều tình huống yêu cầu N_A lớn có thể dẫn đến thời gian mô phỏng rất lâu đặc biệt khi BER thấp (*SNR* cao). Nếu BER nhỏ, lỗi giải điều chế được mô phỏng là hiếm (xác xuất thấp, khả năng xảy ra thấp) nhưng là sự kiện quan trọng. Mục đích của lấy mẫu quan trọng là: *(i)* thay đổi mô phỏng theo cách điều khiển được; *(ii)* tăng số lượng các sự kiện hiếm và quan trọng này trong khi đó vẫn có thể xác định chính xác xuất lỗi giải điều chế. Điều này cần phải thay đổi trong (16.15). Lấy mẫu quan trọng thuộc lớp phương pháp luận mô phỏng được xem là các kỹ thuật giảm phương sai. Mục đích của các kỹ thuật giảm phương sai là triển khai một bộ ước tính không chệch mà biểu lộ phương sai được giảm và/hoặc thời gian mô phỏng được giảm so với mô phỏng Monte Carlo.

Trước khi đề cập việc lấy mẫu quan trọng, lưu ý rằng việc mô tả chi tiết lấy mẫu quan trọng là ngoài phạm vi tài liệu này. Theo đó, thay vì xét ước tính diện tích hình học bằng cách sử dụng các khái niệm cơ bản của lấy mẫu quan trọng.

Trong chương 9 vấn đề mô phỏng Monte Carlo được đề cập bằng cách triển khai bộ ước tính cho diện tích có dạng hai chiều (lưu ý rằng vấn đề này dẫn đến bộ ước tính cho giá trị của π). Trong chương này ta lại xét một bộ ước tính cho diện tích có dạng hai chiều (trong trường hợp này là elip) nhằm làm sáng tỏ kỹ thuật mô phỏng lấy mẫu quan trọng. Cuối chương trình bày mô phỏng hệ thống thông tin đơn giản.

16.4.1. Diện tích elip

Giả sử ta muốn dùng các mô phỏng Monte Carlo để ước tính diện tích elip trong mặt phẳng (x, y). Cụ thể, ta khảo sát bài toán tìm diện tích elip có trục chính theo hướng x có độ dài $2\sqrt{2}$ và trục phụ có độ dài 2.

Xem lại các bộ ước tính Monte Carlo

Elip bao gồm tất cả các điểm (x, y) sao cho:

$$x^2 + 2y^2 < 2 \tag{16.16}$$

Diện tích của elip A_e là:

$$A_e = \int_{y=-1}^{1} \int_{x=-\sqrt{2-2y^2}}^{\sqrt{2-2y^2}} dxdy = \pi\sqrt{2} \cong 4,443 \tag{16.17}$$

Bằng cách viết lại tích phân, ta có thể đưa bài toán vào dạng phù hợp với tích phân Monte Carlo hơn và để giải thích lấy mẫu quan trọng. Muốn vậy,

Trước hết, biểu diễn các giới hạn của tích phân bằng một hàm bị tích bằng cách định nghĩa hàm chỉ báo $h_e(x, y)$, sao cho:

$$h_e(x,y) = \begin{cases} 1, & x^2 + 2y^2 \le 2 \quad \text{trong elip} \\ 0, & x^2 + 2y^2 > 2 \quad \text{ngoài elip} \end{cases} \tag{16.18}$$

Dẫn đến:

$$A_e = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h_e(x,y)\,dxdy \tag{16.19}$$

Tiếp theo, xác định vùng biên giới chứa elip và có một vùng dễ tính toán. Diện tích của vùng biên giới này được ký hiệu là $A_{biên}$. Nhân hàm bị tích với phần tử đơn vị có dạng $A_{biên}/A_{biên}$ cho ta:

$$A_e = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h_e(x,y)\, A_{biên} \frac{1}{A_{biên}}\, dxdy \tag{16.20}$$

Sau đó, xác định cặp biến ngẫu nhiên (X, Y) được phân bố đều trên $A_{biên}$. Hàm pdf của cặp biến ngẫu nhiên này là:

$$f_{XY}(x,y) = \begin{cases} 1/A_{biên}, & h_e(x,y) = 1 \\ \text{tùy ý}, & h_e(x,y) = 0 \end{cases} \tag{16.21}$$

Cuối cùng là, kết hợp (16.20) và (16.21):

$$A_e = A_{biên} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_e(x,y) f_{XY}(x,y)\,dx\,dy \tag{16.22}$$

Như ở trong chương 9, bộ ước tính diện tích elip là

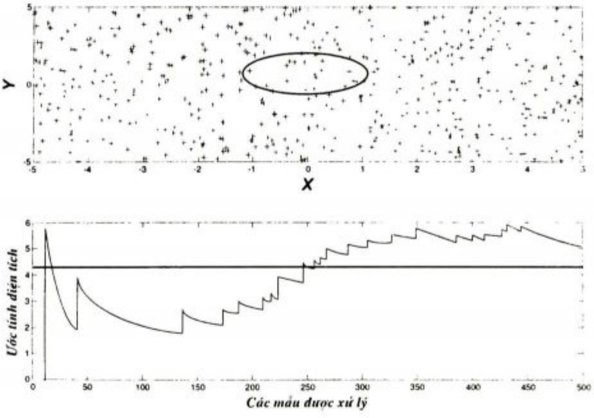
$$\hat{A}_e = A_{biên}\left(\frac{N_e}{N}\right) \tag{16.23}$$

Trong đó N biểu thị số điểm được tạo trong vùng biên giới và N_e biểu thị số các điểm rơi vào trong elip.

Lựa chọn các hộp bao cho các mô phỏng MC

Có thể thấy rằng bộ ước tính được định nghĩa bởi (16.23) là kiên định và không chệch đã tạo ra vùng biên giới đóng bao hoàn toàn elip, được minh hoạ trong ví dụ sau đây

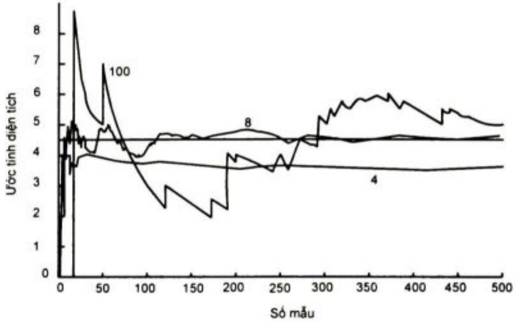
Ví dụ 16.1: Tạo N = 500 điểm trong hộp bao hình vuông được đặt tại trung tâm gốc tọa độ. Sử dụng (16.23) để ước tính diện tích elip với ba kích thước hình biên khác nhau; $A_{biên}$ = 4, 8, và 100. Nửa trên của hình 16.3 cho thấy các điểm mẫu được tạo trong mô phỏng này với $A_{biên}$ = 100 (10 đơn vị trên 1 chiều), và nửa dưới của hình này cho thấy ước tính liên tục của diện tích cùng với giá trị được tính theo lý thuyết. Thấy rõ từ hình 16.3, cần có nhiều hơn 500 mẫu trong mô phỏng này để tạo ước tính diện tích elip chính xác.



Hình 16.3: Quá trình ước tính diện tích với N = 100

Tiếp theo, ta khảo sát ảnh hưởng của việc thay đổi diện tích hình bao biên giới. Đặt $A_{biên}$ = 8 cải thiện đột ngột các thuộc tính hội tụ của bộ ước tính. Kết quả là, vùng được ước tính nằm trong vài phần trăm giá trị đúng chỉ sau 100 mẫu. Dựa vào quan sát này ta có thể kết luận rằng, đường bao biên giới càng nhỏ thì bộ ước tính hội tụ càng nhanh. Nó chính xác tới một điểm. Tuy nhiên, khi $A_{biên}$ = 4, thì bộ ước tính hội tụ nhanh, nhưng tới giá không chính xác. Bộ ước tính bị chệch vì hình bao biên giới không chứa toàn bộ elip nữa. Nếu đường bao biên giới tiếp

tục co lại, thì tốc độ hội tụ tiếp tục được cải thiện nhưng độ chệch tăng và cuối cùng thì ta có thể đơn giản hóa việc tìm diện tích hình bao và không phải là diện tích elip. Các tính chất hội tụ khi $A_{biên} = 100, 8, 4$ được minh họa ở hình 16.4.



Hình 16.4: Minh họa tính hội tụ đối với các diện tích bao khác nhau

Ảnh hưởng của kích thước hình bao biên giới có thể được dự đoán từ (16.23). Mức độ chính xác của tỉ số N_e/N tăng theo N_e. Nói cách khác, ta muốn có nhiều điểm nhất bên trong elip, dẫn đến ta chọn các đường bao nhỏ. Tuy nhiên, bộ ước tính MC sẽ trở nên bị chệch nếu đường bao rơi hoàn toàn vào elip, dẫn đến kích thước đường bao tối ưu là $A_{biên} = 8$, là hình vuông nhỏ nhất bao quanh elip. Thách thức là ở chỗ làm thế nào tìm được các bộ ước tính vừa không bị chệch vừa hội tụ nhanh.

Các vùng biên giới tối ưu

Trong phần mô phỏng trước cho thấy, mô phỏng MC hai chiều sẽ tạo ra các điểm ngẫu nhiên trên một số vùng của mặt phẳng *(x, y)*. Để tạo ra bộ ước tính hội tụ nhanh, ta muốn có bộ ước tính MC sử dụng vùng nhỏ nhất có thể mà vẫn đảm bảo diện tích quan tâm được bao hoàn toàn. Để đơn giản, ban đầu ta giả sử vùng biên giới là hình chữ nhật. Tuy nhiên, không gặp khó khăn khi triển khai thuật toán ước tính theo yêu cầu này. Thuật toán có thể áp dụng cho bất kỳ vùng biên giới của vùng đã cho. Chẳng hạn, xét một biên giới hình chữ nhật. Hình chữ nhật nhỏ nhất chứa elip có diện tích là $4\sqrt{2}$. Vì nó nhỏ hơn diện tích biên giới hình vuông nhỏ nhất, nên bộ ước tính dựa vào vùng biên giới hình chữ nhật sẽ hội tụ nhanh hơn bộ ước tính dựa vào vùng biên giới hình vuông.

Tại đây ra có thể mở rộng tìm hình bao biên giới tốt hơn cho các đa giác bậc cao hơn. Ta có thể tìm các đa giác bao quanh elip và có các diện tích thậm chí nhỏ hơn $4\sqrt{2}$. Sẽ khó tính toán vùng đa giác biên giới theo cách giải tích hơn và cũng sẽ khó tạo các điểm phân bố đều trong đa giác hơn. Vùng biên giới cuối cùng chính là elip. Trường hợp này $A_{biên} = A_e$, và mọi điểm ngẫu nhiên được tạo ra sẽ rơi vào bên trong elip. Vì vậy, $N_e = N$. Phương trình (16.23) chỉ ra rằng mô phỏng sẽ hiệu quả và sẽ tạo ra *kết quả chính xác sau khi tạo ra một điểm ngẫu nhiên*. Tất nhiên, trong trường hợp lấy giới hạn này không cần thiết thực hiện mô phỏng, vì A_e phải được xác định theo cách giải tích trước khi triển khai mã mô phỏng. Dĩ nhiên, nó loại bỏ sự cần thiết mô phỏng.

Một trong những thách thức của việc tạo ra những mô phỏng hiệu quả cao là lỗi tính toán nhỏ có thể gây ra một phần elip rơi ra ngoài vùng biên giới, tạo ra một bộ ước tính bị chệch. Nếu một khi không biết câu trả lời chính xác, thì không thể phát hiện sự chệch bằng cách chỉ quan trắc các kết quả mô phỏng.

Tóm tắt lại khi sử dụng pdf phân bố đều trong các mô phỏng MC, thì nên:

1. Xác định diện tích vùng biên giới theo cách giải tích.

2. Chứng minh theo cách giải tích rằng vùng biên giới bao quanh hoàn toàn diện tích quan tâm.

3. Tìm một giải thuật để tạo các điểm phân bố đều một cách hiệu quả trong vùng biên giới.

4. Giảm thiểu diện tích vùng biên giới, được cho trong các ràng buộc 1, 2, và 3.

5. Chạy mô phỏng đủ lâu để quan sát số lượng lớn các mẫu trong elip.

Các quan tâm và các yêu cầu này rất dễ để hiểu đối với bài toán đơn giản này. Chúng sẽ xuất hiện lại trong các phần sau trong nhiều chiến lược mô phỏng phức tạp hơn, và mức độ phức tạp toán học sẽ làm cho nó khó hình dung và đánh giá đúng hơn.

Các hàm pdf không đều và các hàm trọng số

Tại đây ta mới chỉ xét thuật toán Monte Carlo cơ bản. Các mô phỏng Monte Carlo từ chương 9 ta có hai bộ đếm, một bộ *đếm sự kiện* và một bộ *đếm mẫu*. Bộ đếm mẫu được tăng mỗi lần thử nghiệm ngẫu nhiên được thực hiện (ví dụ tạo một mẫu và kiểm tra xem nếu nó rơi vào vùng quan tâm). Bộ đếm sự kiện được tăng mỗi lần xuất hiện sự kiện quan tâm như xuất hiện một mẫu rơi vào vùng đã cho. Các bộ đếm thường được tăng bằng cách tăng nội dung bộ nhớ. Ta thấy rằng, bằng cách cộng trọng số ω_i vào bộ đếm sự kiện chứ không phải là số 1. *Khái niệm làm tăng các trọng số chứ không phải là làm tăng các số nguyên là trung tâm để áp dụng phương pháp luận lấy mẫu quan trọng cho ước tính BER.*

Trong phương trình (16.19) ta đã nhân hàm bị tích với phần tử đơn vị có dạng $A_{biên}/A_{biên}$ để có được (16.20). Bộ ước tính tổng quát hơn được thiết lập bằng cách nhân hàm bị tích trong (16.19) với *hàm trọng số hai chiều* được chia bởi chính nó $\omega(x,y)/\omega(x,y)$. Giống như trước đó, ta thay $1/\omega(x,y)$ bởi pdf. Cụ thể là:

$$f_{XY}(x,y) = \begin{cases} 1/\omega(x,y), & h_e(x,y)=1 \\ \text{tùy ý}, & h_e(x,y)=0 \end{cases}$$

Dẫn đến:

$$A_e = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_e(x,y).\omega(x,y).f_{XY}(x,y)\,dxdy \qquad (16.24)$$

Đây là dạng tổng quát hơn của (16.22), trong đó $f_{XY}(x,y)$ có thể là pdf bất kỳ khác không trong elip. Bộ ước tính diện tích tương ứng với (16.24) là:

$$\hat{A}_e = \frac{1}{N}\sum_{i=1}^{N} h_e(x_i,y_i).\omega(x_i,y_i) \qquad (16.25)$$

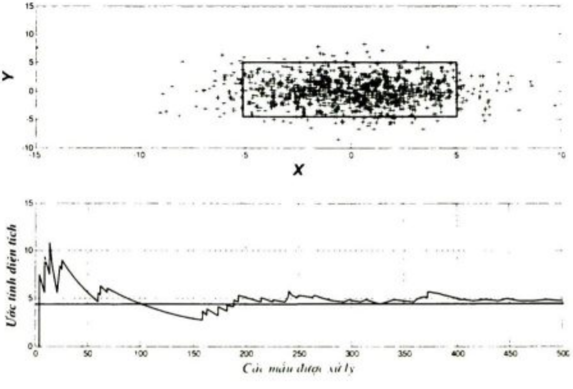
Trong đó các véc tơ ngẫu nhiên (x_i, y_i) là phân bố đồng dạng và độc lập *iid* có pdf là $f_{XY}(x, y)$. Đây là sự tổng quát hoá của kết quả trước, vì với trường hợp phân bố đều, thì đảo của pdf, nghĩa là hàm trọng lượng $A_{biên}$ là không đổi đối với tất cả các điểm bên trong elip. Phương trình (16.25) cho phép ta sử dụng gần như pdf bất kỳ trong mô phỏng và bao gồm pdf phân bố đều như trường hợp đặc biệt. Phương sai và độ chệch của bộ ước tính là hàm của pdf được chọn cho mô phỏng.

Để tránh độ chệch bộ ước tính, pdf phân bố đều phải chứa toàn bộ elip. Nói cách khác, pdf phân bố đều phải khác không trong bên trong elip. Ta có thể thấy rằng, bộ ước tính diện tích không bị chệch với giả thiết $f_{XY}(x, y) > 0$ với tất cả các điểm bên trong elip. Vì có nhiều pdf là khác không trên toàn bộ mặt phẳng (x, y), nên không khó để tạo các bộ ước tính không chệch. Giảm thiểu phương sai là một bài toán khó hơn vì tốc độ hội tụ mô phỏng là một hàm của pdf phức tạp hơn. Ví dụ sau đây làm sáng tỏ bài toán này.

Ví dụ 16.2: Trong ví dụ này ta chỉ xét các pdf phân bố Gausơ và khảo sát tốc độ hội tụ cho ba giá trị độ lệch chuẩn khác nhau là $\sigma = 10, \sigma = 1$ và $\sigma = 0,2$.

Trường hợp $\sigma = 10$

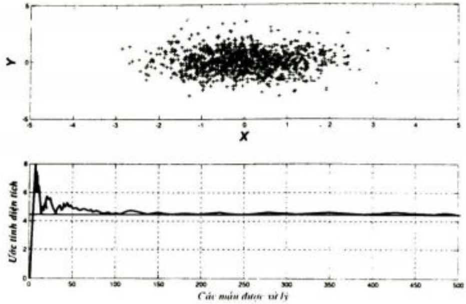
Hình 16.4 cho thấy kết quả với $\sigma = 10$. (Lưu ý hộp bao ban đầu có kích thước 5×5. Elip được che bởi các điểm mẫu). Khi σ lớn, pdf bên trong elip là nhỏ nhưng gần như không đổi. Vì vậy, trọng số tương đối lớn được gán cho mỗi mẫu rơi vào trong elip. Cho thấy một vài bước nhảy lớn sau các khoảng có tốc độ phân rã hyperbolic *(1/N)*. Ước tính diện tích hội tụ đến giá trị đúng *chậm*, trong phần lớn các cách tương tự như pdf phân bố đều và diện tích biên giới lớn.



Hình 16.5: Hiệu năng bộ ước tính với lấy mẫu Gausơ và $\sigma = 10$

Trường hợp $\sigma = 1$

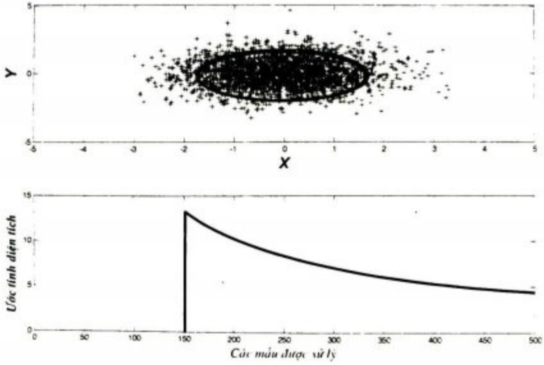
Khi này độ lệch chuẩn đóng gần kích thước của elip và kết quả là tốc độ hội tụ nhanh hơn nhiều. Một số lượng lớn các mẫu rơi vào trong elip, và mỗi mẫu được ấn định một trọng số tương đối nhỏ. Trường hợp này tương ứng với pdf phân bố đều khi vùng biên giới chỉ lớn hơn elip không đáng kể. Kết quả này được thể hiện trong hình 16.6.



Hình 16.6: Hiệu năng bộ ước tính với lấy mẫu Gausơ và $\sigma = 1$

Trường hợp $\sigma = 0{,}15$

Ảnh hưởng của tính cách sự kiện cực trị đã quen thuộc với ta từ các nghiên cứu trước. Ví dụ đơn giản giả sử: ***(i)*** hệ thống truyền tin số có BER bằng 10^{-6}; ***(ii)*** thực hiện mô phỏng MC; ***(iii)*** thu được 99 ký hiệu đầu tiên không có lỗi. Bộ ước tính BER $\hat{P}_E = N_e / N$ bằng 0 khi $1 \leq N \leq 99$ là giá trị quá nhỏ. Nếu ký hiệu số 100 bị lỗi (sự kiện cực trị) thì $\hat{P}_E$ nhảy từ 0 lên 10^{-2} (là quá lớn). Nếu kéo dài các quyết định chính xác theo lỗi này, thì $\hat{P}_E$ sẽ giảm là *1/N* khi *N*>100. Tính cách này sẽ tiếp tục cho tới khi xảy ra lỗi tiếp theo. Về cơ bản giống như tính cách trong hình 16.7 chỉ khác là lỗi trong hình 16.7 được nhân với một trọng số lớn ω_i. Lưu ý rằng các sự kiện cực trị xuất hiện sớm trong chạy mô phỏng có ảnh hưởng rõ ràng hơn các sự kiện cực trị xuất hiện sau.



Hình 16.7: Hiệu năng bộ ước tính với lấy mẫu Gausơ và $\sigma = 0{,}15$

Cả ba mô phỏng đều dùng cùng số gốc ngẫu nhiên. Cả ba bộ ước tính đều không bị chệch và kiên định, nhưng cách thức hội tụ khác nhau rất nhiều. Các cách thức này được tổng hợp trong bảng 16.1.

Bảng 16.1: Cách thức hội tụ

Độ chệch chuẩn σ	Số các mẫu trong Elip	Cách thức hội tụ
Quá nhỏ	Gần như tất cả	Chậm (biểu hiện các sự kiện cực trị)
Phù hợp	Từ 10% đến 90%	Nhanh
Quá lớn	Rất ít	Chậm

Trong ví dụ rất đơn giản này, có rất nhiều phương pháp để dự đoán cách thức hội tụ. Các biểu đồ tán xạ cho biết khi pdf bao phủ quá ít elip hoặc quá nhiều mặt phẳng. Bộ chỉ thị rõ nét khác là hàm trọng số. Hàm trọng số lý tưởng sẽ nhỏ trên toàn bộ diện tích elip. Nếu tập hợp đủ dữ liệu, thì các ước tính và các mẫu của các hàm lỗi cũng cho thấy rõ các cách thức. Về lý tưởng, ta mong muốn thiết lập các khoảng tin cậy cho các bộ ước tính này. Đây là việc rất khó vì các ước tính không có phân bố nhị thức nữa. Có thể lấy gần đúng Gausơ nếu số lượng đủ lớn các ký hiệu được xử lý. Số lượng mẫu cần thiết cho phép lấy gần đúng được chính xác là phụ thuộc vào cách thức hội tụ. Khi hội tụ nhanh chỉ cần vài trăm mẫu. Các bộ ước tính hội tụ chậm tương tự với trường hợp đều và cần vài trăm điểm bên trong elip. Trường hợp cuối cùng, cần quan sát hàng trăm sự kiện cực trị trước khi lấy gần đúng Gausơ.

16.4.2. Tính nhạy cảm với pdf

Việc tìm diện tích elip là bài toán tất định. Các biến ngẫu nhiên được sử dụng trong mô phỏng MC là *hoàn toàn giả định*, và không mô hình hóa cho bất kỳ thiết bị vật lý hoặc tín hiệu nào. Hầu hết các hệ thống truyền thông vật lý đều phải chịu các xáo trộn ngẫu nhiên và xử lý số liệu ngẫu nhiên. Khuynh hướng đầu tiên là triển khai mô phỏng sử dụng cùng hàm pdf nguồn tạp âm làm hệ thống vật lý. Tuy nhiên, ta sẽ thấy là không cần thiết. Mô phỏng có thể sử dụng nguồn tạp âm *bị chệch*, có hàm pdf khác với hàm pdf được dùng trong hệ thống vật lý. Ví dụ, giả sử hệ thống vật lý tạo ra một cặp biến ngẫu nhiên Gausơ *(x, y)* có pdf là:

$$f_{\text{vật lý}}(x,y)=\frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-r^2}}e^{-\frac{1}{2(1-r^2)}\left(\frac{x^2}{\sigma_x^2}-\frac{2rxy}{\sigma_x\sigma_y}+\frac{x^2}{\sigma_y^2}\right)} \tag{16.26}$$

Trong đó σ_x và σ_y là các độ lệch chuẩn của *X* và *Y*, và *r* là hệ số tương quan. Xác xuất mà cặp biến ngẫu nhiên rơi vào trong elip là:

$$B_e=\int_{y=-1}^{1}\int_{x=-\sqrt{2-2y^2}}^{\sqrt{2-2y^2}} f_{\text{vật lý}}(x,y)\,dxdy=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_e(x,y)f_{\text{vật lý}}(x,y)\,dxdy \tag{16.27}$$

Biểu thức này khó ước lượng. Giải pháp MC đơn giản cho bài toán này sử dụng tổng:

$$\hat{B}_e=\frac{1}{N}\sum_{i=1}^{N}h_e(x_i,y_i)=\frac{N_e}{N} \tag{16.28}$$

Trong đó các véc tơ ngẫu nhiên (x_i, y_i) là phân bố đồng nhất độc lập thống kê ***idd*** có pdf là $f_{vật\ lý}(x, y)$. Giả sử ta không có bộ tạo số ngẫu nhiên để tạo ra $f_{vật\ lý}(x, y)$, nhưng ta đã có bộ tạo số ngẫu nhiên để tạo ra các mẫu có phân bố là $f_{mô\ phỏng}(x, y)$. Ta vẫn có thể giải quyết bài toán bằng các kỹ thuật Monte Carlo. Trong các phần trước, ta nhân hàm bị tích của (16.27) với phần tử đơn vị có dạng $\omega(x,y)/\omega(x,y)$. Khác với triển khai trước đây, hàm trọng số bây giờ được định nghĩa là:

$$\omega(x,y)=\begin{cases} f_{\text{vật lý}}(x,y)/f_{\text{mô phỏng}}(x,y), & h_e(x,y)=1 \\ \text{tùy ý}, & h_e(x,y)=0 \end{cases} \tag{16.29}$$

Thay (16.29) vào (16.27) ta được:

$$B_e=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_e(x,y)\,\omega(x,y)\,f_{\text{mô phỏng}}(x,y)\,dxdy \tag{16.30}$$

Có thể được xấp xỉ bởi tổng:

$$\hat{B}_e=\frac{1}{N}\sum_{i=1}^{N} h_e(x_i,y_i)\,\omega(x_i,y_i) \tag{16.31}$$

Các vec tơ ngẫu nhiên (x_i, y_i) là phân bố đồng nhất độc lập thống kê ***idd*** có $f_{mô\ phỏng}(x, y)$. Cũng giống như các bài toán trước, ta phải quan tâm độ chệch, tính kiên định và cách thức hội tụ của bộ ước tính này. Dễ dàng thấy rằng khi $f_{mô\ phỏng}(x, y)>0$ với mọi điểm trong đó $h_e(x,y).f_{\text{vật lý}}(x,y)>0$, thì ước tính sẽ không bị chệch. Cũng cho thấy rằng điều này đảm bảo ước tính là kiên định. Trong bài toán đơn giản hơn, cách thức hội tụ có thể chậm, nhanh, hoặc rất chậm do tồn tại các sự kiện cực trị. Cách thức sẽ phụ thuộc vào hàm trọng số $w(x,y)=f_{\text{vật lý}}(x,y)/f_{\text{mô phỏng}}(x,y)$ bên trong elip. Các tốc độ hội tụ được tổng kết trong bảng 16.2.

Bảng 16.2: Tóm tắt các tốc độ hội tụ

Các PDF bên trong Elip	Hàm trọng số bên trong Elip	Tốc độ hội tụ
$f_{\text{mô phỏng}}(x,y)=f_{\text{vật lý}}(x,y)$	$\omega(x,y)=1$	Giống MC đơn giản
$f_{\text{mô phỏng}}(x,y)>f_{\text{vật lý}}(x,y)$	$\omega(x,y)<1$	Nhanh hơn MC đơn giản
$f_{\text{mô phỏng}}(x,y)<f_{\text{vật lý}}(x,y)$	$\omega(x,y)>1$	Chậm hơn MC đơn giản

16.4.3. Xem xét đầy đủ

Trước khi ứng dụng bài toán ước tính diện tích elip vào hệ thống truyền tin, ta cần phải xem xét đầy đủ cuối cùng. Kênh tạo ra tạp âm, nhưng tạp âm này có thể bị biến đổi bởi máy thu thành phân bố mới có pdf chưa biết. Ta phải tìm xác xuất các mẫu được tạo bởi pdf mới này rơi vào bên trong vùng quan tâm. Bây giờ chỉ làm việc với các pdf hai chiều, giả sử rằng kênh tạo ra tạp âm vật lý (x, y), có pdf là $f_{vật\ lý}(x,y)$. Các biến ngẫu nhiên này được đưa vào máy thu, áp dụng toán tử $g(x,y)=(\alpha,\beta)$. Tập các biến ngẫu nhiên mới này có hàm pdf là $f_{\text{thu}}(\alpha,\beta)$. Xác xuất biến ngẫu nhiên (α,β) rơi vào một số vùng quan tâm, chẳng hạn như vùng elip là:

$$B_e = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_e(\alpha,\beta).f_{thu}(\alpha,\beta).d\alpha\, d\beta \tag{16.32}$$

Tính phức tạp của hàm *g* làm cho việc tính $f_{thu}(\alpha,\beta)$ từ $f_{vật\ lý}(x,y)$ không thực tế. Mô phỏng MC của bài toán này sẽ là:

$$\hat{B}_e = \frac{1}{N}\sum_{i=1}^{N} h_e(\alpha_i,\beta_i) \tag{16.33}$$

Trong đó (α_i,β_i) có pdf là $f_{thu}(\alpha,\beta)$. Có thể chạy mô phỏng này ngay cả khi ta không biết cách tính $f_{rec}(\alpha,\beta)$. Nếu ta tạo các vectơ ngẫu nhiên (x_i, y_i) có pdf là $f_{vật\ lý}(x, y)$ và sau đó cho mỗi véc tơ mẫu qua hàm *g*, thì véc tơ kết quả (α_i,β_i) sẽ có pdf là $f_{thu}(\alpha,\beta)$. Chi tiết hóa biểu thức (16.33) hơn nữa ta có:

$$\hat{B}_e = \frac{1}{N}\sum_{i=1}^{N} h_e\big(g(x_i,y_i)\big) \tag{16.34}$$

Giả sử tại đây ta thay $f_{vật\ lý}(x, y)$ bằng $f_{chệch}(x, y)$. Điều này sẽ làm $f_{thu}(\alpha,\beta)$ thành $f_{thu\text{-}chệch}(\alpha,\beta)$. Để loại bỏ các ảnh hưởng độ chệch này từ ước tính MC, ta cần tính hàm trọng số:

$$\omega(\alpha_i,\beta_i) = \frac{f_{thu}(\alpha_i,\beta_i)}{f_{thu\text{-}chệch}(\alpha_i,\beta_i)} = \frac{f_{vật\ lý}(x_i,y_i)}{f_{chệch}(x_i,y_i)} \tag{16.35}$$

Trong đó $(\alpha_i,\beta_i) = g(x_i,y_i)$. Hàm trọng số này được sử dụng để biến đổi (16.34) nhằm tạo:

$$\hat{B}_e = \frac{1}{N}\sum_{i=1}^{N} \omega\big(g(x_i,y_i)\big)\cdot h_e\big(g(x_i,y_i)\big) \tag{16.36}$$

Có thể được thấy là ước tính không bị chệch và kiên định.

16.4.4. Bài toán truyền thông

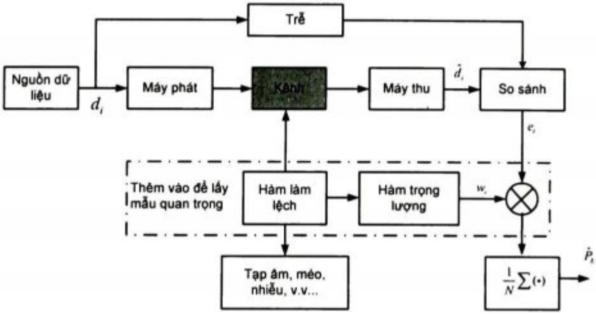
Bài toán hình học tìm diện tích elip có thể ánh xạ thành bài toán tìm BER của hệ thống truyền thông. Hai tập biến ngẫu nhiên *(x, y)* và (α,β) biểu diễn dạng sóng tạp âm kênh và số đo quyết định. Trong bài toán truyền thông, tạp âm kênh thường có nhiều hơn hai chiều, mặc dù số đo quyết định chỉ có một chiều. Quan hệ giữa hai cặp biến ngẫu nhiên bởi toán tử *g(.)*, thể hiện cho máy thu. Tại đây, elip của bài toán hình học tương ứng với các giá trị của số đo quyết định làm cho máy thu mắc lỗi giải điều chế. Việc tìm BER của hệ thống tương đương với việc tìm xác xuất của một mẫu được tạo ra một cách ngẫu nhiên rơi vào bên trong elip. Ta sẽ làm lệch tạp âm kênh để tăng tần xuất của các sự kiện lỗi, và tính hàm trọng số:

$$\omega_i = \frac{f_{vật\ lý}(n_i)}{f_{chệch}(n_i)} \tag{16.37}$$

Bộ ước tính BER là:

$$\hat{P}_e = \frac{1}{N}\sum_{i=1}^{N} \omega(n_i)\, h_e\big(g(x_i,y_i)\big) \tag{16.38}$$

Trong bài toán hình học, ta đã đảm bảo một bộ ước tính không chệch bằng cách làm cho vùng biên giới chứa toàn bộ elip. Ở đây chỉ cần tạo ra tất cả các giá trị của số đo quyết định. Không khó để đáp ứng yêu cầu này. Tốc độ hội tụ trong bài toán hình học phụ thuộc vào cách chọn phương sai nguồn tạp âm thiết lập các điểm mẫu. Nếu nhiều mẫu nằm ngoài elip, thì tốc độ hội tụ chậm. Nếu ta tạo gần như không có mẫu nào nằm ngoài elip, thì tính hội tụ lại chậm do các sự kiện cực trị. Về lý tưởng, ta mong muốn một hỗn hợp các mâu cả bên trong và bên ngoài elip. Trong bài toán hệ thống truyền thông, ta muốn một hỗn hợp các giá trị số đo quyết định cả trong vùng lỗi và vùng không có lỗi. Tập trung quá nhiều mẫu ở một trong hai vùng sẽ làm cho tốc độ hội tụ rất chậm. Hình 16.8 minh hoạ hệ thống với yêu cầu thêm các hàm phụ để thực hiện lấy mẫu quan trọng được biểu diễn bởi các đường kẻ đậm.



Hình 16.8: Mô phỏng hệ thống với việc lấy mẫu quan trọng

16.4.5. Lấy mẫu quan trọng truyền thống và cải tiến

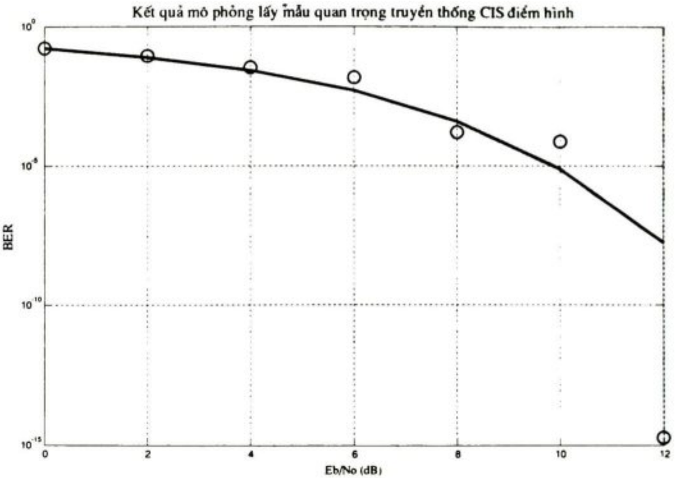
Tồn tại nhiều phương pháp để làm lệch tạp âm khi áp dụng lấy mẫu quan trọng. Hai phương pháp phổ biến nhất được gọi là lấy mẫu quan trọng truyền thống (CIS) và lấy mẫu quan trọng được cải tiến (IIS). Trong CIS ta chỉ tăng phương sai tạp âm kênh, tương đương với hệ thống hoạt động tại SNR thấp. Trong IIS, thay đổi giá trị trung bình của tạp âm chứa không phải là phương sai.

Ví dụ 16.3: Trong ví dụ này ta xét CIS được áp dụng cho hệ thống QPSK vi sai, đã được xét trong chương 10. Mã chương trình Matlab mô phỏng CIS được cho bởi file **NVD16_CISQPSK.m** trong Phục lục 16A.

Hàm này gần giống với mô phỏng Monte Carlo cho hệ thống QPSK được xét trong chương 10 chỉ khác là tạp âm kênh bị lệch bằng cách tăng phương sai tạp âm.

Bộ ước tính BER trong ví dụ này được định nghĩa bởi (16.38). Kết quả điển hình từ mô phỏng này được cho ở hình 16.9. Các vòng trong hình này trình bày các ước tính CIS của BER trong khi đó đường cong nét liền là hiệu năng lý thuyết. Thấy rõ tại SNR cao sự lệch giữa lý thuyết và mô phỏng. Tuy nhiên, các kết quả này chỉ dựa vào 1000 ký hiệu giải điều chế. Sử dụng các mô phỏng MC truyền thống sẽ không thực tế để đo BER bất kỳ nhỏ hơn 10^{-3}.

Tuy nhiên, từ các kết quả mô phỏng CIS cho ta một số ý tưởng về BER đối với các tỷ số lỗi đạt tới 10^{-8}.



Hình 16.9: Minh họa kết quả mô phỏng CIS điển hình

Chương 17
MÔ PHỎNG HỆ THỐNG VÔ TUYẾN TẾ BÀO

17.1. Mở đầu

Rất nhiều hệ thống truyền thông vô tuyến được triển khai để cung cấp truy cập cơ sở hạ tầng truyền thông cho người dùng cố định hoặc di động với tính đa dạng môi trường hoạt động. Hầu hết đều dựa trên khái niệm vô tuyến tế bào. Các hệ thống truyền thông tế bào cho phép nhiều người dùng di động liên lạc đồng thời và liên tục tới các modem vô tuyến tại các trạm gốc BS, chiếm dụng phổ tần RF hữu hạn. Tín hiệu RF thu tại các BS từ mỗi MS được chuyển thành tín hiệu băng tần cơ sở, chuyển tới các trung tâm chuyển mạch di động MSC, MSC kết nối tới mạng điện thoại công cộng PSTN. Tương tự, truyền tin từ PSTN tới BS, truyền tới MS. Các hệ thống tế bào dùng các kỹ thuật: *(i)* đa truy cập phân chia theo tần số FDMA; *(ii)* đa truy cập phân chia theo thời gian TDMA; *(iii)* đa truy cập phân chia theo mã CDMA; *(iv)* đa truy cập phân chia theo không gian SDMA.

Các tuyến truyền thông vô tuyến phải chịu ảnh hưởng của các đặc tính kênh vật lý như: đa đường thay đổi theo thời gian, che chắn do các vật cản lớn trên đường truyền sóng.v.v. Hơn nữa, hiệu năng các hệ thống tế bào có xu hướng bị hạn chế bởi nhiễu từ các người dùng khác, vì thế cần phải có kỹ thuật để mô hình hóa nhiễu một cách chính xác. Việc miêu tả các điều kiện kênh phức tạp này bằng mô hình giải tích đơn giản là rất khó mặc dù tồn tại một số mô hình khả giải tích khá phù hợp với dữ liệu kênh đo được. Tuy nhiên, ngay cả khi kênh được mô hình hóa theo cách phân tích, thì phần lớn các tình huống thật là khó hoặc không thể xây dựng các giải pháp phân tích cho hiệu năng tuyến nối *khi* mã hoá kiểm soát lỗi, cân bằng, phân tập, các mô hình mạng được *nhân tố hóa vào* mô hình tuyến nối. Vì vậy, khi phân tích hiệu năng các tuyến truyền thông tế bào thường cần đến giải pháp mô phỏng.

Tương tự như các tuyến nối không dây, hiệu năng của *hệ thống* vô tuyến tế bào hầu hết được mô hình hiệu quả bằng cách sử dụng mô phỏng (do sự khó khăn khi mô hình hóa số lượng lớn các sự kiện ngẫu nhiên theo thời gian và không gian). Các sự kiện ngẫu nhiên này như: *(i)* vị trí người dùng; *(ii)* số lượng người dùng đồng thời trong hệ thống; *(iii)* các điều kiện truyền sóng; *(iv)* nhiễu; *(v)* thiết lập mức công suất cho mỗi người dùng; *(vi)* nhu cầu lưu lượng của mỗi người dùng, tất cả kết hợp với nhau gây ảnh hưởng lên toàn bộ hiệu năng của một người dùng trong hệ thống tế bào. Các nhân tố trên chỉ là một minh chứng tiêu biểu của nhiều cơ chế vật lý quan trọng để công bố hiệu năng tức thời của một người dùng riêng biệt tại thời điểm nào đó trong hệ thống. Vì vậy, *thuật ngữ hệ thống vô tuyến tế bào, được xem là toàn bộ mật độ người dùng di động và trạm gốc ở khắp vùng phục vụ địa lý* (khác với tuyến kết nối đơn, kết nối một người dùng di động với *một* trạm gốc). Để thiết kế hiệu năng mức hệ thống, cần phải xét tính phức tạp của nhiều người dùng đồng thời trên vùng phủ. Vì vậy, cần phải xét các ảnh hưởng nhiều người dùng lên bất kỳ một trong các tuyến cụ thể giữa BS và MS.

Hiệu năng tuyến nối là hiện tượng phạm vi hẹp (*small-scale phenomenon*) liên quan với những những thay đổi tức thì trong kênh trên một nội vùng nhỏ, hoặc khoảng thời gian nhỏ trên đó công suất thu trung bình được coi là không đổi. Có thể cảm nhận được những giả định này khi thiết kế các mã kiểm soát lỗi, các bộ cân bằng, và các thành phần khác nhằm giảm bớt những ảnh hưởng tức thì của kênh. Tuy nhiên, để xác định toàn bộ hiệu năng hệ thống của một số lượng lớn người dùng trải khắp trên một vùng địa lý rộng, thì cần phải kết hợp chặt chẽ các ảnh hưởng phạm vi rộng (*large-scale*) như: tính cách thống kê của nhiễu; mức tín hiệu phải chựu ảnh hưởng bởi các người dùng riêng biệt trên các khoảng cách lớn (khi này bỏ qua các đặc tính kênh tức thì). Có thể coi mô phỏng mức tuyến nối như là điều chỉnh hiệu năng của hệ thống truyền thông, và mô phỏng mức hệ thống như là một phép tính xấp xỉ sơ bộ chất lượng toàn bộ mà bất cứ người dùng nào được mong đợi tại bất kỳ thời điểm nào.

Các hệ thống tế bào đạt được dung lượng cao (đáp ứng số lượng lớn người dùng) bằng cách cho phép các MS chia sẻ, nghĩa là tái sử dụng kênh truyền thông ở các vùng khác trong vùng phục vụ. Tái sử dụng kênh dẫn đến *nhiễu đồng kênh* giữa những người dùng cùng kênh, được xem là một trong những nhân tố chính làm hạn chế hiệu năng và dung lượng hệ thống tế bào. Vì vậy, cần phải hiểu rõ ảnh hưởng của nhiễu đồng kênh lên dung lượng và hiệu năng khi triển khai các hệ thống tế bào, hoặc khi phân tích và thiết kế các phương pháp luận hệ thống nhằm giảm bớt các ảnh hưởng của nhiễu đồng kênh. Các ảnh hưởng này phụ thuộc vào nhiều nhân tố, như số người dùng chia sẻ kênh và vị trí của họ. Các nhân tố khác liên quan tới kênh truyền sóng nhiều hơn như: *(i)* suy hao tuyến; *(ii)* pha đinh che chắn; *(iii)* mẫu bức xạ ăng ten *vì* các ảnh hưởng này cũng thay đổi theo vị trí của người dùng. *Chương này bàn về việc ứng dụng mô phỏng mức hệ thống trong phân tích hiệu năng hệ thống truyền thông tế bào dưới các ảnh hưởng của nhiễu đồng kênh.* Ta sẽ phân tích một hệ thống tế bào đa người dùng gồm có các ảnh hưởng truyền sóng và ăng ten của một hệ thống điển hình. Mặc dù hệ thống được xét là đơn giản, nhưng việc phân tích có thể dễ dàng được mở rộng cho các tính năng khác của hệ thống tế bào.

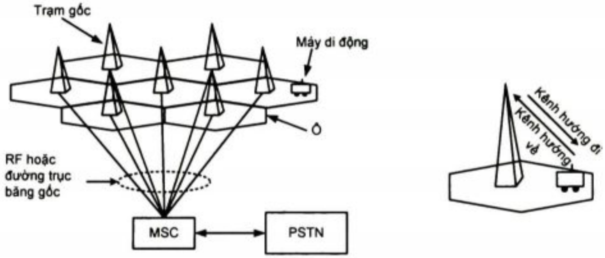
Theo đó, chương này sẽ trình bày cơ bản tổng quan các nhân tố quan trọng để mô phỏng hiệu năng tổng thể một hệ thống truyền tin vô tuyến. Một số tham số thiết kế như: *(i)* hệ số tái sử dụng tần số tế bào (kích cỡ cụm); *(ii)* xác suất nghẽn; *(iii)* tỉ số tín hiệu trên nhiễu SIR trung bình; *(iv)* độ rộng búp ăng ten, tất cả đều ảnh hưởng lên hiệu năng hệ thống truyền thông vô tuyến tổng thể. Hơn nữa, chúng không độc lập nhau mà quan hệ mật thiết nhau và thường được dùng để dung hòa hiệu năng hệ thống. Chẳng hạn, kích cỡ cụm tế bào nhỏ sẽ làm tăng dung lượng kênh khả dụng của hệ thống nhưng làm gia tăng mức nhiễu của mỗi người dùng. Tương tự, việc phân đoạn làm giảm nhiễu đồng kênh nhưng lại làm giảm dung lượng hệ thống. Do tính đối nghịch nên cần phải mô phỏng để tìm giải pháp dung hòa. Mức nhiễu đồng kênh do những người dùng không mong muốn thiết lập nền tạp âm hoạt động và được thấy bởi một người dùng di động tại một vị trí xác định, và chương này cũng sẽ minh họa mô hình hóa và mô phỏng hiệu quả các ảnh hưởng của các tham số thiết kế hệ thống. Chiến lược mô phỏng là tạo ra một phân tích Monte Carlo của nhiều chớp ảnh về hiệu năng của hệ thống được nhận biết bởi nhiều người dùng di động hoạt động tại các vị trí ngẫu nhiên, tập hợp các chớp ảnh đó được dùng để tạo số liệu thống kê hiệu năng hoạt động trên các vùng phủ mong muốn. Việc mô phỏng hệ thống đòi hỏi phải mô hình hóa cẩn thận các hiệu ứng không gian của người dùng, các mẫu bức xạ ăng

ten cụ thể, việc cấp phát các kênh vô tuyến trên không gian và các mức nhiễu phạm vi rộng trên không gian đó. Chương cũng sẽ trình bày một số ví dụ và phương pháp luận để làm sáng tỏ các vấn đề quan trọng này nhằm thiết kế và mô phỏng các hệ thống truyền thông vô tuyến.

17.2. Hệ thống vô tuyến tế bào

17.2.1. Mô tả mức hệ thống

Các hệ thống tế bào tạo ra vùng phủ sóng khắp vùng phục vụ địa lý bằng cách chia vùng địa lý thành các phần được gọi là các tế bào như được cho ở hình 17.1. Phổ tần khả dụng cũng được chia thành các nhóm kênh để phân bổ cho ô. Các trạm gốc được định vị trong mỗi ô, được trang bị các modem vô tuyến để liên lạc với người dùng di động. Các kênh tần số vô tuyến RF được dùng ở hướng từ BS đến MS được gọi là các *kênh hướng đi* (đường xuống), ngược lại là các *kênh hướng về* (đường lên). Các kênh đường lên và đường xuống cùng nhau nhận biết một kênh tế bào song công. Khi dùng song công phân chia theo tần số FDD, thì các kênh đường lên và đường xuống được phân tách tần số. Khi sử dụng song công phân chia theo thời gian TDD, thì các kênh đường lên và đường xuống sử dụng cùng tần số nhưng dùng các khe thời gian khác nhau để truyền.



Hình 17.1: Kiến trúc cơ bản của một hệ thống truyền thông tế bào

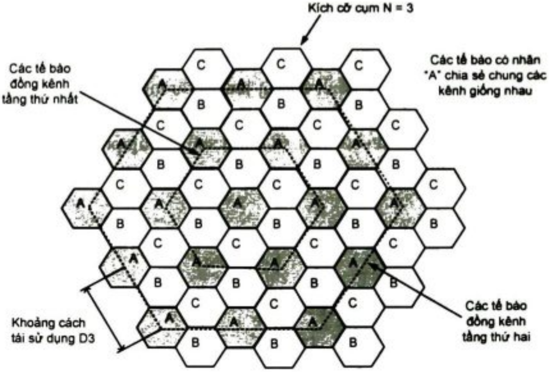
Các hệ thống tế bào dung lượng lớn tận dụng khả năng tái sử dụng tần số giữa các ô. Dẫn đến các tế bào đồng kênh (các tế bào chia sẻ cùng tần số) phải đủ xa nhau để giảm thiểu nhiễu đồng kênh. Tái sử dụng kênh bằng cách bao phủ vùng phục vụ bằng các cụm N ô, như được thấy ở hình 17.2, trong đó N là kích cỡ cụm.

Phổ tần RF khả dụng cho vùng phục vụ địa lý được phân bổ cho mỗi cụm sao cho các tế bào trong một cụm không chia sẻ bất cứ kênh nào. *Nếu: **(i)*** M kênh hợp lại thành toàn bộ phổ tần khả dụng cho vùng phục vụ; ***(ii)*** sự phân bố các người dùng là đồng đều trên vùng phục vụ, *thì* mỗitế bàođược ấn định M/N kênh. Khi các cụm được tái tạo trên vùng phục vụ, thì việc tái sử dụng kênh dẫn đến xếp tầng cácté bào đồng kênh, và nhiễu đồng kênh sẽ sinh ra từ truyền lan năng lượng RF giữa các trạm gốc đồng kênh và các người dùng di động. Ví dụ, xảy ra nhiễu đồng kênh trong một hệ thống tế bào khi MS đồng thời thu được các tín hiệu từ trạm gốc trong tế bào của nó, cũng như từ các trạm gốc đồng kênh các tế bào từ các tầng lân cận. Trong ví dụ này, một tuyến kết nối đường xuống đồng kênh (BS đến MS) là tín hiệu mong muốn, và các tín hiệu đồng kênh khác được thu bởi MS tạo thành nhiễu đồng kênh tổng tại máy thu. Mức công

suất nhiễu đồng kênh liên quan mật thiết với khoảng cách phân tách giữa các tế bào đồng kênh. Nếu mô hình hóa các tế bào bằng hình lục giác (6 cạnh) như trong hình 17.2, thì khoảng cách nhỏ nhất giữa tâm của hai tế bào đồng kênh được gọi là *khoảng cách tái sử dụng* D_N là:

$$D_N = \sqrt{3N}R \tag{17.1}$$

Trong đó R là bán kính cực đại của tế bào (hình 6 cạnh được vẽ nội tiếp trong vòng tròn bán kính). Vì vậy, thấy rõ từ hình 17.2 khi kích cỡ cụm giảm (khoảng cách tái sử dụng D_N giảm), thì nhiễu giữa các tế bào đồng kênh tăng.



Hình 17.2 Cụm hóa các tế bào: Mô tả mẫu tái sử dụng ba ô

Tại một thời điểm bất kỳ, mức nhiễu đồng kênh phụ thuộc vào số các tế bào đồng kênh. Như đã đề cập, các tế bào đồng kênh được nhóm thành các *tầng*. Số các tế bào đồng kênh trong một tầng phụ thuộc vào bậc tầng và dạng hình học được chọn để trình bày tế bào (ví dụ, vùng phủ của một trạm gốc cụ thể). Với dạng lục giác, các tế bào đồng kênh gần nhau nhất được định vị trong tầng thứ nhất và có 6 tế bào đồng kênh. Tầng thứ hai gồm 12 tế bào đồng kênh, tầng thứ ba có 18 tế bào và cứ tiếp tục như thế. Theo đó nhiễu đồng kênh tổng là tổng các tín hiệu nhiễu đồng kênh từ tất cả các tế bào đồng kênh của tất cả các tầng. Tuy nhiên, các tế bào đồng kênh thuộc tầng thứ nhất có ảnh hưởng mạnh hơn, vì chúng gần với tế bào nơi mà nhiễu được đo.

Nhiễu đồng kênh được coi là một trong các nhân tố chính làm hạn chế chất lượng tuyến nối và dung lượng hệ thống, và có vai trò quan trọng trong việc dung hòa giữa dung lượng hệ thống (vấn đề phạm vi rộng) và chất lượng tuyến nối (vấn đề phạm vi hẹp). Chẳng hạn, giải pháp để đạt được dung lượng lớn mà không tăng phổ tần RF được cấp phát là giảm khoảng cách tái sử dụng kênh bằng cách giảm kích cỡ cụm N. Tuy nhiên, giảm kích cỡ cụm sẽ tăng nhiễu đồng kênh và làm giảm chất lượng kết nối.

Mức nhiễu tại thời điểm nào đó là ngẫu nhiên, và phải được mô phỏng bằng cách mô hình hóa cho: *(i)* môi trường truyền sóng RF giữa các ô; *(ii)* định vị vị trí của những người dùng

di động. Ngoài ra, tính cách thống kê lưu lượng của mỗi người dùng và kiểu phân bổ kênh tại trạm gốc sẽ xác định mức nhiễu tức thời và dung lượng của hệ thống.

Ta có thể ước tính ảnh hưởng của nhiễu đồng kênh bằng *tỉ số tín hiệu trên nhiễu SIR* của tuyến truyền thông, được định nghĩa là tỉ số giữa công suất tín hiệu mong muốn S trên công suất nhiễu tổng I. Vì S và I đều là các biến ngẫu nhiên (do ảnh hưởng truyền lan sóng RF, tính di động của người dùng và tính thay đổi lưu lượng), nên SIR cũng là biến ngẫu nhiên. Cho nên, mức độ ảnh hưởng nghiêm trọng của nhiễu đồng kênh lên hiệu năng hệ thống thường được phân tích ở dạng xác suất ngừng hoạt động hệ thống, trong trường hợp cụ thể này được định nghĩa là xác xuất mà SIR nhỏ hơn mức ngưỡng SIR_0 cho trước, nghĩa là:

$$P_{\text{ngừng hoạt động}} = \Pr\left[SIR < SIR_0\right] = \int_0^{SIR_0} p_{SIR}(x)dx \tag{17.2}$$

Trong đó $p_{SIR}(x)$ là hàm mật độ xác xuất pdf của SIR. Lưu ý rằng, sự khác biệt giữa định nghĩa *xác xuất ngừng tuyến nối* (nhằm phân loại ngừng hoạt động dựa vào giá trị BER cụ thể hoặc ngưỡng E_b/N_0 đối với hiệu năng tiếng chấp nhận được), và *xác xuất ngừng hoạt động hệ thống* (xét ngưỡng SIR cụ thể để hiệu năng di động của một người dùng có thể chấp nhận được).

Các giải pháp giải tích để ước tính xác xuất ngừng hoạt động trong hệ thống tế bào như đã được đề cập ở chương 11, cần có các mô hình có thể xử lý theo cách giải tích cho các ảnh hưởng truyền sóng RF, sự di động của người dùng và sự thay đổi lưu lượng, nhằm có được biểu thức $p_{SIR}(x)$. Đáng tiếc, thật khó sử dụng các mô hình giải tích cho các ảnh hưởng này (do các quan hệ phức tạp của chúng với mức tín hiệu thu). Vì vậy, ước tính xác suất ngừng hoạt động trong hệ thống tế bào thường dựa vào mô phỏng (tạo ra tính mềm dẻo trong ước tính). Chương này sẽ trình bày một mô phỏng để minh họa hệ thống truyền thông tế bào đơn giản nhằm nhấn mạnh các khía cạnh mang tính hệ thống của hệ thống truyền tin bao gồm hiệu năng đa người dùng, thiết kế lưu lượng và tái sử dụng kênh. Để xây dựng mô phỏng mức hệ thống, số lượng các khía cạnh của các kết nối thông tin riêng biệt phải được xem xét gồm mô hình kênh, mẫu phát xạ ăng ten và quan hệ giữa E_b/N_0 (chẳng hạn SIR) và hiệu năng khả chấp nhận.

Để mô phỏng một hệ thống tế bào, ta phải lập mô hình toán học các thành phần hệ thống. Phần sau, ta sẽ trình bày việc lập mô hình các thành phần hệ thống.

17.2.2. Mô hình hóa hệ thống vô tuyến tế bào

Ta xét vài khía cạnh hệ thống truyền thông tế bào, chúng sẽ rất hữu hiệu khi triển khai mô hình mô phỏng để nghiên cứu chi tiết hệ thống.

Trung kế và cấp dịch vụ

Ta bắt đầu việc đề cập một số vấn đề dung lượng hệ thống vô tuyến tế bào. Giống như các hệ thống điện thoại cố định, các hệ thống vô tuyến tế bào dựa vào *khái niệm trung kế* để cung cấp dịch vụ truyền thông cho một số lượng lớn người dùng, chiếm một tài nguyên hữu hạn, cụ thể với trường hợp truyền thông tế bào là phổ RF hoặc số lượng các kênh khả dụng. Sở dĩ sử dụng các kỹ thuật trung kế là do đặc tính thống kê của người dùng, được mô tả chủ yếu bằng hai khía cạnh sau:

1. Một người dùng truy cập vào hệ thống nghĩa là *yêu cầu một cuộc gọi*, trên cơ sở ngẫu nhiên trong suốt một khoảng thời gian và khoảng thời gian τ giữa hai yêu cầu gọi liên tiếp từ một người dùng, tuân theo phân bố hàm số mũ. Do vậy, pdf cơ bản là:

$$p_\tau(\tau) = \lambda_u \, e^{-\lambda_u \tau} \tag{17.3}$$

Trong đó λ_u là số yêu cầu cuộc gọi trung bình trên một đơn vị thời gian (các cuộc gọi trên thời gian) của một người dùng. Nếu ta xét mật độ U người dùng, thì sự phân bố của khoảng thời gian giữa hai yêu cầu cuộc gọi liên tiếp được tạo ra bởi hai người dùng bất kỳ cũng theo luật số mũ. Số các yêu cầu cuộc gọi trung bình là $\lambda = U.\lambda_u$.

2. Khoảng thời gian cuộc gọi cũng là một biến ngẫu nhiên tuân theo phân bố hàm mũ sao cho các cuộc gọi ngắn có khả năng xuất hiện nhiều hơn các cuộc gọi dài. Nếu ký hiệu khoảng thời gian cuộc gọi là s, thì hàm mật độ xác suất pdf của s là:

$$f_S(s) = \mu . e^{-\mu s} \tag{17.4}$$

Trong đó $\frac{1}{\mu} = H$ là khoảng thời gian gọi trung bình (các đơn vị thời gian)

Dựa vào tính cách thống kê này, rất nhiều người dùng có thể cùng chia sẻ một số lượng nhỏ các kênh trong một nhóm các kênh. Với mỗi BS trong hệ thống tế bào tạo ra một nhóm C kênh trung kế khả dụng cho mọi người dùng trong vùng phủ của trạm gốc BS. Do nhu cầu một người dùng không truy cập vào hệ thống tại mọi thời điểm, nên ta có thể cấp phát các kênh cho các người dùng. Khi cuộc gọi được kết thúc, thì kênh trở về nhóm các kênh khả dụng. Tuy nhiên, do thiếu kênh vô tuyến khả dụng ở trạm gốc, nên không phải lúc nào một người dùng cũng được phép thiết lập tuyến nối vô tuyến với các BS phục vụ. Khi này, tất cả các kênh bận phục vụ người dùng khác, và yêu cầu cuộc gọi đó bị *nghẽn*. Dựa vào tính cách thống kê của các người dùng, số lượng kênh khả dụng trong nhóm, và các đặc tính chính xác của hệ thống trung kế, ta có thể xác định xác xuất một người dùng có nhu cầu gọi bị nghẽn (do thiếu kênh rỗi). Xác xuất này thường được gọi là xác xuất nghẽn, là số đo "*cấp phục vụ*" của một hệ thống trung kế. Tính cách thống kê của một người dùng có thể được tổng kết bằng lưu lượng A_u được tạo ra bởi người dùng này, có đơn vị là Erlang và được xác định là:

$$A_u = \lambda_u H \tag{17.5}$$

Đối với một hệ thống gồm U người dùng, thì lưu lượng cung cấp tổng (đơn vị Erlang) là:

$$A = UA_u = \lambda H \tag{17.6}$$

Đặc tính quan trọng của hệ thống trung kế là công bố chất lượng dịch vụ cho người dùng biết hệ thống xử lý cuộc gọi bị nghẽn như thế nào. Tồn tại hai chiến lược cơ bản: ***(i)*** Các yêu cầu cuộc gọi không tìm được các kênh rỗi khả dụng bị nghẽn và được xoá. Khi này hệ thống trung kế được xem là *xóa các cuộc gọi nghẽn*. ***(ii)*** Các cuộc gọi nghẽn được giữ trong một hàng đợi và được phục vụ ngay khi kênh trở nên khả dụng. Khi này các hệ thống trung được gọi là *trễ các cuộc gọi nghẽn*. Ta sẽ tập trung vào hệ thống trung kế *xoá cuộc gọi nghẽn* vì hệ thống này thường gặp trong thực tế.

Vì vậy, ta giả sử rằng các cuộc gọi nghẽn được xoá. Ngoài ra, có các giả thiết sau đây:

- Cuộc gọi đến là không nhớ. Nói cách khác, người dùng bất kỳ (kể cả các người dùng có các cuộc gọi trước bị nghẽn) đều có thể yêu cầu một cuộc gọi tại bất cứ thời điểm nào.

- Có một số vô hạn người dùng.

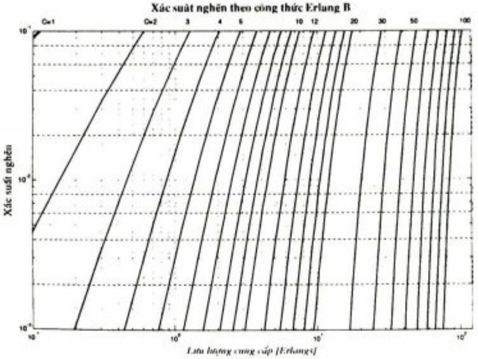
- Có C kênh trung kế khả dụng trong nhóm.

Dưới các điều kiện này *xác suất một cuộc gọi bị nghẽn* P_B được cho bởi công thức Erlang B. Công thức Erlang B quan hệ hóa giữa số lượng kênh trung kế C, xác xuất nghẽn P_B và lưu lượng A, có thể hoặc là *lưu lượng cung cấp* hoặc *lưu lượng mang*. Trường lưu lượng cung cấp, P_B là xác xuất nghẽn phải chịu bởi mật độ người dùng tạo ra lưu lượng A, *cung cấp* cho hệ thống trung kế với C kênh. Trường lưu lượng mang, A là lưu lượng mang tối đa bởi hệ thống trung kế chứa C kênh tại xác suất nghẽn P_B. Lưu lượng mang bởi hệ thống trung kế cũng là một phép đo dung lượng của hệ thống. Công thức Erlang B là:

$$P_B = \frac{A^C / C!}{\sum_{k=0}^{C} A^k / k!} \qquad (17.7)$$

Khi cho trước lưu lượng cung cấp và số các kênh trung kế, thì xác xuất nghẽn được tính toán bởi mã chương trình Matlab **NVD17_erlang_b.m** trong Phục lục 17A.

Đồ thị Erlang B với số lượng kênh trung kế làm tham số được cho ở hình 17.3, khi này số lượng kênh trung kế C được thấy dọc bên trên đồ thị. Mỗi khi chạy chương trình **NVD17_erlangb.m** chương trình này sẽ gọi hàm **NVD17_erlang_b(A,c)**. Cho trước công thức Erlang B, thì các ảnh hưởng của kích cỡ cụm lên dung lượng hệ thống ở dạng số lượng người dùng được ước lượng.



Hình 17.3: Xác suất nghẽn được cho bởi công thức Erlang B với số trung kế làm tham số

Ví dụ 17.1: Xét một hệ thống tế bào trong đó 400 cặp kênh đường lên và đường xuống khả dụng cho toàn bộ hệ thống. Mỗi tế bào có bán kính 5 km, và các trạm gốc được trang bị các ăng ten vô hướng được định vị tại tâm của mỗi ô. Các giả định gồm: *(i)* mỗi người dùng tạo

ra lưu lượng 0,02 Erlang và kích cỡ cụm $N = 7$; ***(ii)*** người dùng được phân bố đồng đều khắp vùng phục vụ, mỗi tế bào được cấp phát $N_C = 400/7 \approx 57$ kênh. Ngoài ra, nếu coi các cuộc gọi nghẽn được xoá và xác xuất nghẽn $P_B = 0,02$ là chấp nhận được, thì lưu lượng tối đa A_C được mang bởi mỗi tế bào được cho bởi công thức Erlang B. Kết quả là:

$$A_C = 46,8 \quad \text{Erlang/tế bào}$$

Dễ dàng tính được số lượng người dùng trên mỗi tế bào. Vì $A_u = 0,02$ Erlang trên mỗi người dùng, nên số lượng người dùng trên mỗi tế bào là:

$$U = \frac{A_C}{A_u} = 2340 \text{ người dùng/tế bào}$$

Tại đây giảm kích cỡ cụm $N = 3$ không thay đổi vùng phủ của mỗi tế bào, theo đó mỗi tế bào được cấp $N_C = 400/3 = 133$ kênh tại $P_B = 0,02$. Cho ta:

$$A_C = 120,1 \text{ Erlangs tế bào}$$

Đối với lưu lượng mang tối đa trên mỗi tế bào. Cho ta:

$$U = \frac{A_C}{A_u} = \frac{120,1}{0,02} = 6005 \text{ người dùng/tế bào}$$

Ta thấy rằng, bằng cách giảm kích cỡ cụm ta đạt được dung lượng cao hơn (do tăng cường tái sử dụng và hiệu quả trung kế). Tuy nhiên, các tế bào đồng kênh trong kích cỡ cụm $N = 3$ gần nhau hơn so với trường hợp kích cỡ cụm $N = 7$. Sử dụng (17.1) ta có:

$$D_3 = 15 \text{ km} \quad \text{với } N = 3$$

$$D_7 = 22,9 \text{ km} \quad \text{với } N = 7.$$

Thấy rõ, mức nhiễu sẽ lớn hơn khi kích cỡ cụm $N = 3$ so với kích cỡ cụm $N = 7$. Vì vậy, kích cỡ cụm $N = 3$ thể hiện chất lượng kết nối thấp hơn.

Mô hình kênh

Khi phân tích hiệu năng hệ thống tế bào, cần phải mô hình hóa chính xác các ảnh hưởng truyền lan sóng vô tuyến lên tín hiệu thu, chúng là những nhân tố chính làm suy thoái hiệu năng hệ thống. Như được trình bày trong chương 14, việc đặc tính hóa thống kê các tín hiệu thu (tín hiệu mong muốn và nhiễu) liên quan đến hai ảnh hưởng truyền lan sóng cơ bản: ***(i)*** phadinh phạm vi hẹp do đa đường trong một nội vùng; ***(ii)*** che chắn (pha đinh phạm vi rộng) gây ra bởi các suy hao ngẫu nhiên của tín hiệu trung bình cục bộ như cây cối, công trình xây dựng, địa hình. Các kết quả đo cho thấy, mức tín hiệu trung bình cục bộ có thể được mô hình hóa khá chính xác như một biến ngẫu nhiên loga chuẩn. Khi biểu diễn theo đơn vị dB, mức tín hiệu trung bình cục bộ theo sự biến thiên chuẩn hóa, và được đặc tính hóa bởi một giá trị trung bình vùng và độ lệch chuẩn theo đơn vị dB. Giá trị trung bình *vùng* là một hàm của: ***(i)*** khoảng cách phát thu (T-R); ***(ii)*** mức công suất phát; ***(iii)*** độ lợi ăng ten, trong khi đó độ lệch chuẩn che chắn phụ thuộc vào môi trường vật lý. Khi thiết kế hệ thống hoặc mô phỏng tổng quát, cần phải xét các ảnh hưởng của pha đinh phạm vi hẹp và che chắn, mặc dù trong nhiều trường hợp che chắn tín hiệu mong muốn và nhiễu là nguyên nhân chính gây suy thoái hiệu năng. Chẳng hạn, các kỹ thuật phân tập không gian, trải phổ, đan xen, mã hoá kênh được triển khai rộng rãi để đối phó

các ảnh hưởng của pha đinh phạm vi hẹp, vì vậy tín hiệu thu chủ yếu phụ thuộc vào sự thay đổi kênh phạm vi rộng. Ở đây, để đơn giản coi rằng, các ảnh hưởng pha đinh phạm vi hẹp được lấy trung bình và chỉ có che chắn và tổn hao đường truyền được xem xét. Tuy nhiên, các ảnh hưởng pha đinh phạm vi hẹp có thể dễ dàng được kết hợp trong phân tích.

Giả sử các ảnh hưởng pha đinh phạm vi hẹp được lấy trung bình, mức công suất trung bình cục bộ của tín hiệu mong muốn hoặc tín hiệu nhiễu ở đây được ký hiệu là ρ, phải chịu biến thiên chuẩn *lognormal*. Mức công suất trung bình cục bộ được mô hình hóa là:

$$X = 10\log_{10}\rho = m_X + \chi \qquad dBW \tag{17.8}$$

Trong đó m_X là mức công suất trung bình vùng dBW (hoặc tổn hao đường truyền sóng phạm vi rộng trung bình dB), χ là một biến ngẫu nhiên phân bố Gausơ trung bình không dB có độ lệch chuẩn σ_X dB do che chắn gây ra bởi chướng ngại vật lớn. Công suất trung bình vùng m_X thường được mô hình là hàm của: *(i)* khoảng cách phát thu (T-R) *d*; *(ii)* số mũ suy hao đường truyền sóng γ; *(iii)* công suất phát P_T dBW; *(iv)* độ lợi đại ăng ten phát và thu G_R, G_T đơn vị dB. Cụ thể:

$$m_X = P_T + G_T(\theta_T,\phi_T) + G_R(\theta_R,\phi_R) - 10K\gamma\log_{10}d \quad dBW \tag{17.9}$$

Hằng số *K* trong (17.9) bao gồm tất cả các số hạng không thay đổi trong mô hình. Các góc θ_T và ϕ_T là góc ngẩng và góc phương vị của tín hiệu phát hướng đến máy thu trong khi đó θ_R và ϕ_R là góc ngẩng và góc phương vị của các tín hiệu đến ăng ten thu. Các góc θ_T, ϕ_T, θ_R và ϕ_R phụ thuộc vào các vị trí tương đối của trạm gốc và các ăng ten di động.

Ví dụ 17.2: Xét một trạm MS đang thu tín hiệu từ một BS như được minh họa ở hình 17.4(a). Các giả thiết cơ bản: *(i)* khoảng cách phát thu *d* = 1200 *m*; *(ii)* tuyến truyền thông giữa BS và MS là nhìn thẳng LOS; *(iii)* các hệ toạ độ được cho trong hình. Xác định các góc phương vị và góc ngẩng tại MS và BS:

Góc phương vị

Từ hình vẽ ta xác định được góc phương vị của tín hiệu được phát bởi BS (xem hình 17.4(b)) là:

$$\phi_T = \cos\left(\frac{663,3}{1200}\right) = 56,4^o$$

Tương tự, góc phương vị của tín hiệu đến tại ăng ten MS là:

$$\phi_R = 236,4^o$$

Góc ngẩng

Coi rằng ăng ten trạm BS cao 30 *m* và người dùng giữ điện thoại di động cách mặt đất là 1,5 m như được cho ở hình 17.4(a). Từ hình vẽ, góc ngẩng của tín hiệu phát bởi BS hướng đến MS là:

$$\theta_T = 180^0 - \tan\left(\frac{1200}{30-1,5}\right) = 91,6^0$$

Tương tự cho góc ngẩng của tín hiệu đến tại ăng ten MS, ta có:

$$\theta_R \approx 88,4^0$$

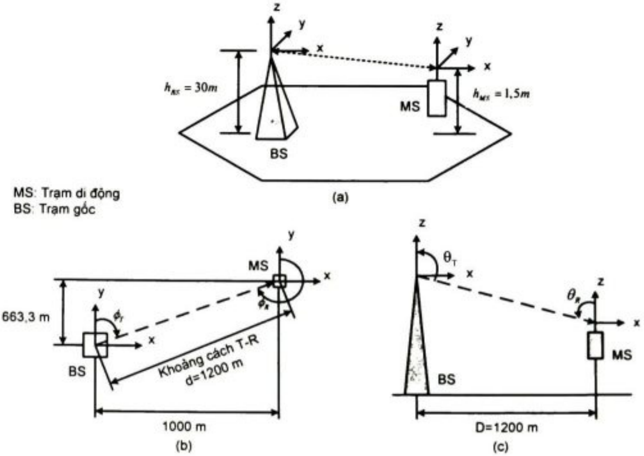
Cho thấy, các góc ngẩng và phương vị phụ thuộc vào các vị trí tương đối của các ăng ten thu phát, cũng như hệ toạ độ được sử dụng. Khi mô phỏng hệ thống tế bào, các hệ tọa độ định rõ các góc và các khoảng cách phải được xác định cẩn thận.

Một kết luận quan trọng nữa từ ví dụ trên là: với các hệ thống tế bào lớn (*macro-cellular*) ở đó bán kính tế bào lớn hơn 1 km, các khoảng cách phát thu d lớn hơn nhiều so với hiệu số chiều cao của các ăng ten BS và ăng ten di động thì ta có thể coi rằng:

$$\theta_T \approx \theta_R \approx 90^0 \tag{17.10}$$

Và bỏ qua θ_T và θ_R trong (17.9). Ta có:

$$\begin{aligned} G_T(\theta_T,\phi_T) &\Rightarrow G_T(\phi_T) \\ G_R(\theta_R,\phi_R) &\Rightarrow G_R(\phi_R) \end{aligned} \tag{17.11}$$



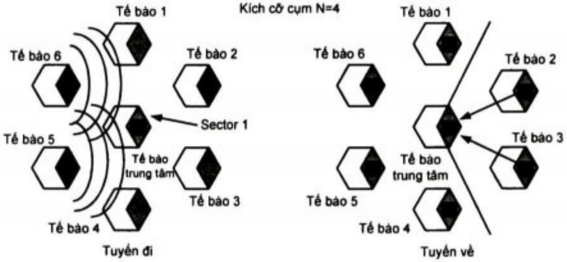
Hình 17.4: Các góc khởi hành và góc đến

Phân đoạn tế bào

Để giảm nhiễu đồng kênh, các hệ thống truyền thông tế bào thường sử dụng ăng ten phân đoạn tại trạm gốc. Mỗi ăng ten phân đoạn phát xạ bên trong một sector cụ thể của tế bào, và mỗi sector được cấp phát một tập con các kênh khả dụng. Vì vậy, nhờ tính hướng của các ăng

ten BS giảm được nhiễu đồng kênh tổng tại ăng ten thu (tại trạm gốc hoặc di động), được sáng tỏ trong ví dụ dưới đây.

Ví dụ 17.3: Xét một hệ thống tế bào với các trạm gốc được trang bị các ăng ten phân đoạn trên cả hai tuyến. Giả sử độ rộng búp sóng $BW = 120^o$ và *tỉ số trước-sau* của ăng ten phân đoạn là vô cùng sao cho không có công suất được phát xạ bên ngoài độ rộng búp sóng. Cũng coi rằng, dùng kích cỡ cụm $N = 4$. *Trên tuyến đi* (đường xuống), số các trạm gốc gây nhiễu trong tầng thứ nhất giảm từ 06 xuống 02 (các BS tại các tế bào 5 và 6) như được thấy ở hình 17.5 là nhờ tính hướng của các ăng ten trạm BS. *Trên tuyến về* (đường lên), số MS đồng kênh trong tầng thứ nhất gây nhiễu BS tại tế bào trung tâm cũng giảm từ 06 xuống 02 (MS tại các tế bào 2 và 3). *Vì vậy, việc phân đoạn là cách giảm nhiễu đồng kênh hiệu quả.* Thấy rõ, mức độ giảm nhiễu đồng kênh phụ thuộc vào độ rộng búp sóng ăng ten phân đoạn và kích cỡ cụm được sử dụng.



Hình 17.5: Các ăng ten phân đoạn trên cả tuyến đi và tuyến về

Các ăng ten phân đoạn được sử dụng trong thực tế có *tỉ số trước sau hữu hạn*. Kết quả là, số lượng tín hiệu nhiễu đồng kênh từ các tế bào đồng kênh trong tầng thứ nhất vẫn là 06, nhưng một số trong chúng rất yếu, do chúng bị suy hao bởi tỉ số trước sau. Hơn nữa, các ăng ten phân đoạn thường được thấy trong thực tế có các độ rộng búp là 120° (3 sector), hoặc 60° (6 sector).

Nhiễu đồng kênh tổng

Dựa vào giả thiết, các tín hiệu nhiễu thu riêng biệt chỉ bị ảnh hưởng bởi che chắn và tổn hao truyền sóng, nên nhiễu đồng kênh tổng được mô hình hóa là sự hợp thành của các tín hiệu nhiễu, các mức công suất trung bình cục bộ của nó theo sự biến thiên *lognormal*. Thường giả thiết rằng, dịch pha trong mỗi tín hiệu nhiễu riêng biệt thay đổi đáng kể do tán xạ, vì thế có thể coi rằng các pha là ngẫu nhiên. Do vậy, nhiễu đồng kênh tổng I tại vị trí cho trước được mô hình hóa là *tổng* các tín hiệu phân bố *lognormal*. Theo đó:

$$I = \sum_{i=1}^{k} I_i, \qquad (17.12)$$

Trong đó I_i khi biểu diễn theo đơn vị dB, được mô hình hóa như trong (17.8).

Thừa nhận rằng phân bố của tổng các biến ngẫu nhiên *lognormal* được xấp xỉ bởi phân bố lognormal khác, và một vài phương pháp đã được đề xuất để tính giá trị trung bình và độ

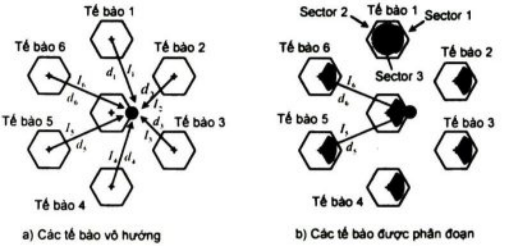
lệch chuẩn theo đơn vị dB. Hai kỹ thuật thông dụng nhất là phương pháp của Wilkinson và phương pháp của Schwartz và Yeh. Một khi biết phân bố tín hiệu nhiễu đồng kênh thành phần I_i, (nói cách khác các giá trị trung bình m_{I_i} và độ lệch chuẩn σ_{I_i} của nhiễu I_i), ta có thể tính giá trị trung bình m_I và độ lệch chuẩn σ_I của nhiễu tổng I bằng phương pháp của Wilkinson hoặc phương pháp của Schwartz và Yeh.

Ảnh hưởng của phân đoạn

Tại đây, ta đã có được cách tính nhiễu đồng kênh, ta phân tích hơn nữa các ảnh hưởng của phân đoạn lên dung lượng và chất lượng kết nối của hệ thống tế bào. Ta biết rằng, việc phân đoạn tế bào làm giảm nhiễu đồng kênh, nhưng cũng làm giảm hiệu suất trung kế, giảm lưu lượng tổng được mang bởi ô. Mỗi sector của tế bào sẽ được cấp phát một tập con các kênh thuộc tập kênh được cấp phát cho ô. Theo lý thuyết lưu lượng, khi chia kênh thành tập con kênh, thì tổng lưu lượng tối đa được mang bởi các tập con luôn luôn nhỏ hơn lưu lượng tối đa được mang bởi toàn bộ kênh.

Ví dụ 17.4: Xét hệ thống tế bào AMPS (hệ thống điện thoại di động tiên tiến) sử dụng kích cỡ cụm $N = 4$. Ngoài ra, coi rằng sẵn có 395 cặp kênh lưu lượng đường lên và đường xuống. Trong ví dụ này ta ước tính lưu lượng tối đa được mang bởi mỗi tế bào và nhiễu đồng kênh. Ta xét hai trường hợp sau: *(a)* ăng ten vô hướng tại BS; *(b)* ăng ten phân đoạn tại BS. Ta chỉ phân tích tuyến kết nối hướng đi.

a) Trường hợp ăng ten vô hướng, các trạm gốc sẽ phát xạ mọi hướng với cùng cường độ. Chúng sẽ giao thoa nhau và thu nhiễu từ tất cả các tế bào đồng kênh. Nếu chỉ xét tầng thứ nhất của các tế bào đồng kênh, thì nhiễu đồng kênh tổng tại MS là tổng của tất cả các tín hiệu nhiễu đồng kênh như được thấy ở hình 17.6 (a). Từ hình này, ta có thể tính *SIR* trung bình *vùng* tại MS được định vị tại biên giới của ô. Đây là trường hợp tồi nhất và được cho bởi:



Hình 17.6: Hệ thống tế bào dùng các ăng ten trạm gốc phân đoạn và vô hướng

$$SIR_{\text{trung bình}} = m_S - m_I \quad dB \tag{17.13}$$

Trong đó m_S là công suất trung bình vùng của tín hiệu mong muốn và m_I là công suất trung bình vùng của nhiễu tổng (lưu ý rằng cả hai đều được biểu diễn theo dB).

Ta sử dụng (17.9) và (17.12) để tính m_S và m_I, để đơn giản ta giả sử rằng ăng ten trạm MS là vô hướng ($G_R(\theta_R, \phi_R) = 0\,dB$), $K = 1$ và tất cả các BS phát cùng mức công suất $P_T = 0$dBW. Do đó:

$$m_s = -10\log_{10} R \tag{17.14}$$

$$m_I = 10\log_{10}\left(\sum_{i=1}^{6} 10^{(-10\gamma\log_{10} d_i)/10}\right) \tag{17.15}$$

Trong đó R là bán kính tế bào, γ là số mũ suy hao đường truyền sóng, d_i, i = 1,..., 6, là khoảng cách phát thu. Sử dụng kích cỡ cụm $N = 4$, giả sử rằng $R = 1000$ và $\gamma = 4$, ta có thể chỉ ra rằng:

$$d_1 = d_4 = 3605{,}55 \ \ m \tag{17.16}$$

$$d_2 = d_3 = 2645{,}75 \ \ m \tag{17.17}$$

$$d_5 = d_6 = 4358{,}90 \ \ m \tag{17.18}$$

Do vậy:

$$m_S = -120{,}0 \qquad dBW \tag{17.19}$$

$$m_I = -132{,}35 \qquad dBW \tag{17.20}$$

Dẫn đến kết quả:

$$SIR_{\text{trung bình}}^{\text{vô hướng}} = 12{,}35 \ dB \tag{17.21}$$

Để phân tích lưu lượng, ta lại giả sử rằng $N = 4$ và người dùng được phân bố đồng đều khắp diện tích tế bào. Mỗi tế bào được cấp phát $N_C = 395/4 \cong 98$ kênh. Giả sử rằng các cuộc gọi bị nghẽn được xoá và xác suất nghẽn $P_B = 0{,}02$ thì lưu lượng tối đa được mang trên mỗi tế bào (theo công thức Erlang B) là:

$$A_C^{\text{vô hướng}} = 86{,}0 \text{ Erlang/tế bào} \tag{17.22}$$

Nếu một người dùng tạo ra lưu lượng 0,02 Erlang, thì tế bào vô hướng có thể hỗ trợ tới $86{,}0/0{,}02 \cong 4300$ người dùng tại xác suất nghẽn $P_B = 0{,}02$.

b) Tại đây ta giả sử rằng sử dụng phân đoạn 120^0 (3 sector trên tế bào) như được thấy ở hình 17.6(b). Để phân tích nhiễu đồng kênh sử dụng cùng giải pháp ở phần a. Ta giả sử ăng ten lý tưởng có tỉ số trước sau vô hạn sao cho chỉ có các tế bào 5 và 6 gây nhiễu lên tế bào trung tâm. Vì vậy, số lượng tế bào nhiễu đồng kênh trong ví dụ lý tưởng này giảm từ sáu (tất cả các tế bào đồng kênh trong tầng thứ nhất) xuống chỉ còn 2, và:

$$m_I = 10\log_{10}\left(10^{(-10\gamma\log_{10} d_5)/10} + 10^{(-10\gamma\log_{10} d_6)/10}\right) = -142.57 \qquad dBW \tag{17.23}$$

$$m_S = -120{,}0 \qquad dBW \tag{17.24}$$

Cuối cùng:

$$SIR_{\text{trung bình}}^{\text{phân đoạn}} = 22{,}6 \quad dB \tag{17.25}$$

Để phân tích lưu lượng dung lượng, thừa nhận rằng mỗi sector được cấp phát $N_S = 395/(4\times 3) \cong 33$ kênh. Lưu lượng tối đa được mang bởi mỗi sector (giả sử xác xuất nghẽn $P_B = 0{,}02$) được cho bởi:

$$A_S^{\text{phân đoạn}} = 23{,}7 \text{ Erlang/Sector} \tag{17.26}$$

Đối với tế bào 3 sector, con số này được nhân với 3. Do vậy:

$$A_C^{\text{phân đoạn}} = 23{,}7 \times 3 = 71{,}1 \text{ Erlang/tế bào} \tag{17.27}$$

Lại giả sử rằng mỗi người dùng tạo ra lưu lượng 0,02 Erlang, thì tế bào được phân đoạn thành 3 sector có thể hỗ trợ tới $71{,}1/0{,}02 \cong 3550$ người dùng tại xác suất nghẽn $P_B = 0{,}02$.

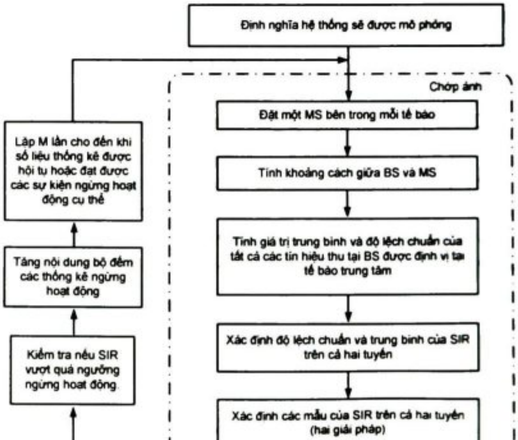
Từ các kết quả này cho thấy, bằng cách phân đoạn tế bào, ta cải thiện được chất lượng kết nối xấp xỉ 10 dB ($\Delta SIR = 22{,}56\text{-}12{,}35 = 10{,}21$ dB) so với trường hợp vô hướng. Tuy nhiên, việc cải thiện chất lượng kết nối đạt được tại mức độ chi phí hiệu suất trung kế, vì vậy lưu lượng tổng được mang trên mỗi tế bào giảm từ 86,0 Erlang/tế bào với ăng ten vô hướng xuống 71,1 Erlang/tế bào với phân đoạn. Sau khi phân đoạn có thể giảm N nếu muốn.

Trong ví dụ trên, ta tính *SIR* cho trường hợp tồi nhất vì giả thiết MS được định vị tại biên giới ô. Rõ ràng rằng tại các vị trí gần trạm gốc phục vụ, *SIR* sẽ lớn hơn. Hạn chế khác ở ví dụ trên là bỏ qua các ảnh hưởng của che chắn. Khi xét đến phân bố không gian của các MS và ảnh hưởng của che chắn, thì *SIR* trở thành biến ngẫu nhiên. Hiệu năng hệ thống tế bào phải được đánh giá bằng xác xuất ngừng hoạt động, được định nghĩa trong (17.2) là xác xuất mà *SIR* thấp hơn mức tối thiểu có thể chấp nhận được SIR_0.

Tiếp theo sẽ trình bày phương pháp luận để mô phỏng hệ thống tế bào đơn giản, xét đến *phân bố không gian* của người dùng và *ảnh hưởng che chắn* lên tín hiệu thu. Các kết quả mô phỏng là *SIR*, cho ta ước tính hiệu năng hệ thống. Mặc dù tính đơn giản của mô phỏng nhưng phương pháp luận có thể được xem là cốt lõi cho các mô phỏng phức tạp hơn, như được mô tả bởi Cardieri và Rappaport.

17.3. Phương pháp luận mô phỏng

Qui trình mô phỏng bao gồm việc mô hình hóa *chớp ảnh* vị trí MS. Tại mỗi chớp ảnh, ta tính toán các số liệu thống kê (trung bình và phương sai) của *SIR* tại BS (*tuyến về*) và tại MS *(tuyến đi)*, ta cũng xét vị trí người dùng và các điều kiện truyền sóng. Mô phỏng một vài chớp ảnh để tạo ra tập mẫu đủ lớn sao cho đạt được các kết quả thống kê hợp lý. Tiến trình mô phỏng được cho ở hình 17.7. Dưới đây ta mô tả một số khía cạnh mô phỏng.



Hình 17.7: Tiến trình ước tính SIR và trạng thái ngừng hoạt động trong hệ thống tế bào sử dụng mô phỏng Monte Carlo

17.3.1. Mô phỏng

Ta chỉ xét tầng thứ nhất của các tế bào đồng kênh. Số sector trong mỗi tế bào là một tham số mô phỏng được chọn là: *(i)* một, tức là ăng ten vô hướng; *(ii)* ba, tức là phân đoạn 120^0; *(iii)* sáu, tức là phân đoạn 60^0. Ta coi rằng, độ lợi đại ăng ten phân đoạn là không đổi bên trong sector và bằng 0 dB, như được thấy ở hình 17.8. Các ăng ten phân đoạn có tỉ số trước-sau bằng B dB, là một tham số mô phỏng.

Khi phân đoạn 120^0, thì các mẫu phát xạ của các ăng ten phân đoạn cho sector 1 được mô hình hóa là:

$$G_T^{(1)}(\phi) = G_R^{(1)}(\phi) = \begin{cases} 0 \;\; dB, & -\pi/3 < \phi \le \pi/3 \\ B \;\; dB, & \text{nếu khác} \end{cases} \tag{17.28}$$

Mẫu phát xạ cho sector 2 và 3 tìm được bằng cách lần lượt quay $G_T^{(1)}(\phi)$ và $G_R^{(1)}(\phi)$ đi 120^0 và 240^0 tương ứng.

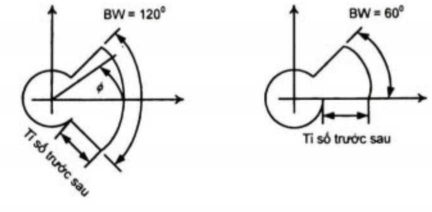
Khi phân đoạn 60^0, ta có:

$$G_T^{(1)}(\phi) = G_R^{(1)}(\phi) = \begin{cases} 0 \;\; dB, & 0 < \phi \le \pi/3 \\ B \;\; dB, & \text{nếu khác} \end{cases} \tag{17.29}$$

Mẫu phát xạ cho các sector 2 đến 6 đạt được bằng cách quay $G_T^{(1)}(\phi)$ và $G_R^{(1)}(\phi)$ đi 60^0. Giả sử cả hai ăng ten trạm gốc đường lên và đường xuống là đồng nhất. Các ăng ten di động được coi là ăng ten vô hướng cho cả đường lên và đường xuống.

Mô hình kênh gồm suy hao đường truyền sóng và che chắn lognormal. Số mũ suy hao truyền sóng và độ lệch chuẩn của che chắn lognormal là các tham số mô phỏng. Mô phỏng bao gồm hai bước cơ bản sau đây:

- *Định nghĩa hệ thống đích sẽ được mô phỏng*
- *Tạo các chớp ảnh của các vị trí di động và tính toán SIR*



Hình 17.8: Mô hình đối với ăng ten phân tế bào khi phân đoạn 120^0 và 60^0

a) Định nghĩa hệ thống đích sẽ được mô phỏng

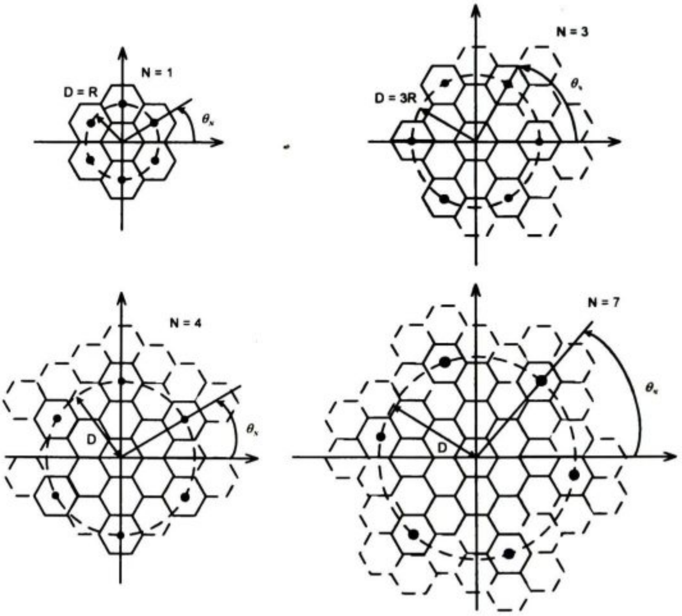
Tại đây ta định nghĩa các đặc tính truyền sóng (các tham số kênh) và vị trí các tế bào đồng kênh.

Các đặc tính truyền sóng (các tham số kênh) gồm:

- *Bán kính tế bào, R*

- *Mũ tổn hao truyền sóng (γ)*
- *Độ lệch chuẩn che chắn lognormal (σ) đơn vị dB*
- *Mức công suất phát trạm gốc ($P_{T,BS}$)*
- *Mức công suất phát MS ($P_{T,MS}$)*
- *Số sector trong mỗi ô*
- *Tỉ số trước-sau của các ăng ten phân đoạn*
- *Số chớp ảnh sẽ được mô phỏng (M)*

Chương trình mô phỏng trên Matlab cho phần này được cho bởi file **NVD17_Intialization.m** ở Phụ lục 17A. Chương trình mô phỏng hoàn chỉnh được cho ở file **NVD17_Cellsim.m** trong Phụ lục 17A.



Hình 17.9: Vị trí của các tế bào đồng kênh hoặc N = 1, N = 3, N = 4, và N = 7

Vị trí các tế bào đồng kênh: Để tiện, ta sẽ chọn hệ toạ độ cực và vuông góc để biểu diễn các vị trí của các trạm gốc và di động trong mô phỏng. Các trạm gốc sẽ được định vị tại tâm của các ô. Trạm gốc tại tế bào trung tâm, ở đó thực hiện đo nhiễu đồng kênh, và tương ứng với tâm các hệ trục toạ độ. Vị trí các tế bào đồng kênh phụ thuộc vào kích cỡ cụm N và bán kính tế bào R. Với tầng thứ nhất, tất cả các tế bào đồng kênh được định vị trên đường tròn bán kính $D = \sqrt{3N}R$, được trung tâm tại tế bào trung tâm, và đồng khoảng cách tới mỗi tế bào như được thấy ở hình

17.9. Góc θ_N (xem bảng 17.1) xác định vị trí góc của tế bào đồng kênh thứ nhất. Bằng phương pháp hình học đơn giản, ta có thể thấy rằng, với kích cỡ cụm *N*, vị trí tế bào đồng kênh thứ *i* là (sử dụng ký hiệu véc tơ):

$$x_{BS_i}\hat{x}+y_{BS_i}\hat{y}=\sqrt{3N}R\left\{\cos\left[\theta_N+(i-1)\tfrac{\pi}{6}\right]\hat{x}+\sin\left[\theta_N+(i-1)\tfrac{\pi}{6}\right]\hat{y}\right\} \quad (17.30)$$

Trong đó $\hat{x}$ và $\hat{y}$ là các véc tơ đơn vị theo phương trục *x* và *y*, tương ứng. Minh họa thực hiện phần mô phỏng bằng đoạn chương trình Matlab sau:

```
% Vị trí của BS (ô trung tâm được định vị tại x=0,y=0)
% Vị trí(góc)tâm của mỗi cụm trong tầng thứ nhất
theta_N          = [ pi/6 0 pi/6 asin(1/(2*sqrt(7)))];
% khoảng cách góc giữa các ô trung tâm của 6 cụm trong tầng thứ nhất
theta            = pi/3*[ 0:5] ';
aux_1            = [ 1 0 2 3 0 0 4];
ind              = aux_1(cluster_size);
% vị trí [ x y] của các ô trung tâm của tất cả các cụm trong tầng thứ nhất
bs_position      = [ sqrt(3*cluster_size)*r_cell*cos(theta +...
   theta_N(ind)) sqrt(3*cluster_size)*r_cell*sin(theta+theta_N(ind))];
```

Bảng 17.1: Quan hệ giữa N, R, và θ_N trong hình 17.9

Kích cỡ cụm *N*	*D*	θ_N
1	*R*	$\frac{\pi}{6}$
3	3*R*	0
4	$2\sqrt{3}R$	$\frac{\pi}{6}$
7	$\sqrt{21}R$	Arctan($2/\sqrt{3}$)

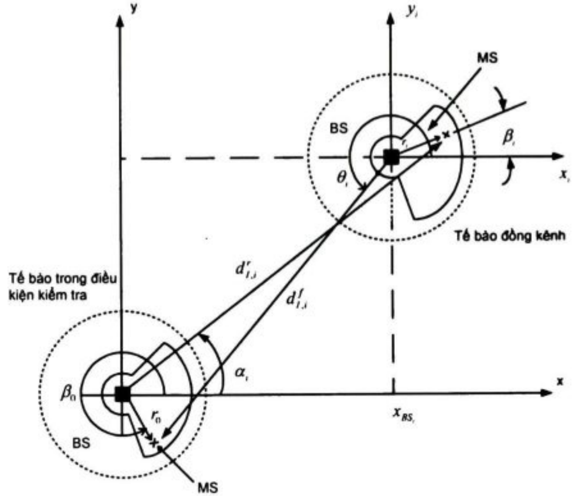
Lưu ý rằng trong đoạn chương trình Matlab trên thì **bs_position(i,1)** và **bs_position(i, 2)** tương ứng với các thành phần trong các phương $\hat{x}$ và $\hat{y}$ của véc tơ biểu diễn vị trí của trạm gốc thứ *i*.

b) Tạo các chớp ảnh của các vị trí MS và tính SIR

Phần mô phỏng này sẽ thực hiện ước lượng Monte Carlo. Tạo các chớp ảnh của các vị trí của các MS chia sẻ cùng kênh, và mỗi chớp ảnh tính toán *SIR*. Phần mô phỏng này được thực hiện *M* lần.

Bước 1: Đặt một MS trong mỗi tế bào. Các MS được giả sử được phân bố đều trên khắp diện tích tế bào. Các sector của một tế bào được cấp phát các tập kênh khác nhau. Do vậy, chỉ xẩy ra nhiễu đồng kênh đối với các sector được cấp phát cùng một tập kênh. Trong mô phỏng, coi rằng các MS ở tế bào trung tâm và ở các tế bào đồng kênh được định vị trong cùng một sector

trong các tế bào của chúng. Sử dụng hệ toạ độ cực, vị trí MS thứ i bên trong tế bào của nó được mô tả bởi: ***(i)*** khoảng cách r_i giữa MS và BS phục vụ; ***(ii)*** góc β_i giữa tham chiếu và phương truyền lan (xem hình 17.10). Lưu ý rằng, β_i và r_i được xác định theo hệ toạ độ được trung tâm tại BS của tế bào thứ *i*. Vì phân bố của vị trí MS là đồng đều trên diện tích ô, nên β_i được phân bố đều trong khoảng $[0, 2\pi]$ và khoảng cách r phân bố theo pdf.



Hình 7.10: Biểu diễn vị trí máy di động

$$f_{R_i}(r_i) = \frac{2r_i}{R^2} \qquad 0 \le r_i \le R \tag{17.31}$$

Lưu ý rằng, để đơn giản trong mô phỏng, ta coi tế bào là hình tròn.

Sector được sử dụng trong một chớp ảnh cụ thể được chọn ngẫu nhiên, với xác suất chọn sector bất kỳ là như nhau:

$$\text{Phân đoạn } 120^0 \Rightarrow \Pr\{\text{sector } i \text{ được chọn}\} = 1/3, \quad i=1, 2, 3$$

$$\text{Phân đoạn } 60^0 \Rightarrow \Pr\{\text{sector } i \text{ được chọn}\} = 1/6, \quad i=1, 2, ..., 6$$

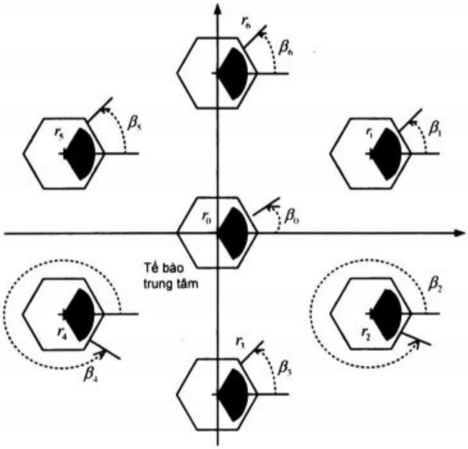
Một khi sector đã được chọn, ta xác định được các góc β_i, cần lưu ý rằng β_i được phân bố đồng đều trong sector đã chọn:

$$\text{Phân đoạn } 120^0 \Rightarrow (2s\text{-}3)\frac{\pi}{3} < \beta_i \le (2s\text{-}1)\frac{\pi}{3}$$

$$\text{Phân đoạn } 60^0 \Rightarrow (s\text{-}1)\frac{\pi}{3} < \beta_i \le s\frac{\pi}{3}$$

Trong đó s là sector đã được chọn ($s = \{1,2,3\}$ cho phân đoạn 120^0, và $s = \{1, ..., 6\}$ cho phân đoạn 60^0). Hình 17.11 minh họa về một chớp ảnh cho phân đoạn 120^0, trong đó các MS được định vị tại sector 1. Thực hiện mô phỏng này bằng đoạn chương trình sau:

```
% Xác định sector được mô phỏng trong chớp ảnh này
% --- chọn (ngẫu nhiên) một sector-------
sector           = unidrnd(num_sectors(sec));
%--- đặt MS mong muốn trong sector đã chọn-------
des_user_beta = rand(1)*phi_BW(sec) + phi_center(sector,sec);
des_user_r  =sqrt(rand(1).*(r_cell^2));
% --- đặt MS đồng kênh trong sector đã chọn của ô đồng kênh---
co_ch_user_beta = rand(6,1)*phi_BW(sec) + phi_center(sector,sec);
co_ch_user_r     = sqrt(rand(6,1))*(r_cell);
```



Hình 7.11: Chớp ảnh đối với phân đoạn 120^0 khi máy di động trong sector 1

Vị trí *(x,y)* của các MS đồng kênh và mong muốn được tính toán theo:

```
des_user_position      = des_user_r*[ cos(des_user_beta) sin(des_user_beta)];
co_ch_user_position  = [ co_ch_user_r.*cos(co_ch_user_beta)
co_ch_user_r.*sin(co_ch_user_beta)] ...
     + bs_position;
```

Bước tiếp theo là xác định các khoảng cách giữa các MS đồng kênh đến BS tại tế bào trung tâm (được dùng để tính toán các giá trị trung bình của các tín hiệu nhiễu trên tuyến về) và

các khoảng giữa MS tại tế bào trung tâm và các BS đồng kênh (được dùng để xác định giá trị trung bình tín hiệu nhiễu trên tuyến đi).

Bước 2: Xác định khoảng cách giữa MS và BS. Để tính toán nhiễu đồng kênh tại BS và MS ở tế bào trung tâm, cần phải xác định

• *Vị trí của các MS đồng kênh theo BS tại tế bào trung tâm*: Sử dụng ký hiệu véc tơ, vị trí của MS thứ i theo BS tại tế bào trung tâm là (xem hình 17.10):

$$d_{I,i}^{r}\left(\cos\alpha_i\hat{\mathbf{x}}+\sin\alpha_i\hat{\mathbf{y}}\right)=r_i\left(\cos\beta_i\hat{\mathbf{x}}+\sin\beta_i\hat{\mathbf{y}}\right)+\left(x_{BS_i}\hat{\mathbf{x}}+y_{BS_i}\hat{\mathbf{y}}\right) \tag{17.32}$$

$d_{I,i}^{r}$ là khoảng cách giữa BS tại tế bào trung tâm và MS thứ *i*, α_i là góc tới của tín hiệu từ MS thứ *i* đập vào ăng ten BS. Vì vậy, $d_{I,i}^{r}$ & α_i là độ dài và hướng của véc tơ được cho bởi vế phải phương trình (17.32).

• *Vị trí của MS tại tế bào trung tâm theo BS đồng kênh thứ i*: (xem hình 17.10):

$$d_{I,i}^{f}\left(\cos\theta_i\hat{x}+\sin\theta_i\hat{y}\right)=r_0\left(\cos\beta_0\hat{x}+\sin\beta_0\hat{y}\right)\ -\ \left(x_{BS_i}\hat{x}+y_{BS_i}\hat{y}\right) \tag{17.33}$$

$d_{I,i}^{f}$ là khoảng cách giữa BS thứ *i* và MS tại tế bào trung tâm, và θ_i là góc xuất phát của tín hiệu phát bởi BS thứ *i* hướng đến MS tại tế bào trung tâm.

Bước 3: Xác định con số thống kê SIR trên cả hai tuyến. Trong bước này, trình bày 3 giải pháp để tính toán *SIR*. Chúng đều xét đến tính thay đổi theo *lognormal* của tín hiệu thu. Theo đó, cần có giá trị trung bình của các tín hiệu nhiễu và tín hiệu mong muốn.

• ***Giá trị trung bình và độ lệch chuẩn tính theo dB của mỗi tín hiệu***

1. Các tín hiệu mong muốn: Sử dụng (17.9), các giá trị trung bình tính theo đơn vị dB của các tín hiệu mong muốn trên tuyến đi và tuyến về lần lượt là

$$m_S^f = P_{T,BS} - 10\gamma\log_{10} r_0 \quad \text{dBW} \tag{17.34}$$

$$m_S^r = P_{T,MS} - 10\gamma\log_{10} r_0 \quad \text{dBW} \tag{17.35}$$

Lưu ý rằng, các độ lợi đại ăng ten BS trên cả 2 tuyến trong các biểu thức này được đặt là 0 dB, vì MS được định vị trong sector của BS. Ngoài ra, độ lợi ăng ten của MS trên cả 2 tuyến cũng đặt là 0 dB vì dùng các ăng ten đẳng hướng. Độ lệch chuẩn của các tín hiệu mong muốn bằng độ lệch chuẩn che chắn. Vì vậy:

$$\sigma_S^f = \sigma_S^r = \sigma$$

2. Các tín hiệu nhiễu: Việc xác định giá trị trung bình của tín hiệu nhiễu tổng trên cả 2 tuyến cần thiết hơn. Như đã được đề cập, nhiễu đồng kênh tổng trên cả 2 tuyến được mô hình hóa là tổng các tín hiệu nhiễu đồng kênh thành phần. Tức là:

$$I^f = \sum_i I_i^f \tag{17.36}$$

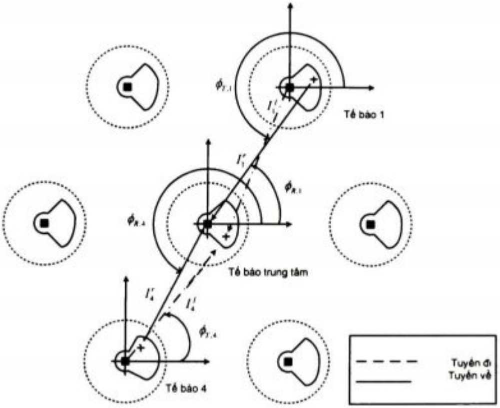
$$I^r = \sum_i I_i^r \tag{17.37}$$

Lần lượt cho tuyến đi và tuyến về. Hình 17.12 cho thấy một chớp ảnh đối với phân đoạn 120°. Mô phỏng 6 tế bào đồng kênh nhưng, chỉ vẽ các tín hiệu từ các tế bào đồng kênh 1 và 4.

Vì trong mô hình này xét các ảnh hưởng của che chắn và suy hao đường truyền nên các tín hiệu nhiễu tổng I^f và I^r là tổng của các tín hiệu phân bố theo hàm loga I_i^f và I_i^r. Như đã đề cập I^f và I^r đều có phân bố *loganormal*. Giá trị trung bình và độ lệch chuẩn tính theo dB của I^f và I^r là các hàm của các giá trị trung bình và các độ lệch chuẩn của các tín hiệu nhiễu và được tính bằng các phương pháp Wilkinsson hay Schwartz và Yeh (được cho ở phụ lục 17C). Các giá trị trung bình tính theo dBW của I^f_i và I^r_i theo công thức (17.9), theo đó ta có:

$$m_{I,i}^f = P_{T,BS} + G_{T,i}(\phi_{T,i}) - 10\gamma \log_{10} d_{I,i}^f \qquad \text{dBW} \tag{17.38}$$

$$m_{I,i}^r = P_{T,MS} + G_{R,0}(\phi_{R,i}) - 10\gamma \log_{10} d_{I,i}^r \qquad \text{dBW} \tag{17.39}$$



Hình 17.12: Minh họa chớp ảnh cho phân đoạn 120^0

Độ lợi ăng ten $G_{T,i}(\phi_{T,i})$ và $G_{R,0}(\phi_{R,i})$ phụ thuộc vào vị trí tương đối của MS và được mô tả theo (17.28) và (17.29). Lưu ý rằng, ta giả sử dùng ăng ten đẳng hướng trên cả 2 tuyến. Độ lệch chuẩn của tín hiệu nhiễu bằng với độ lệch chuẩn che chắn σ:

$$\sigma_{I,i}^f = \sigma_{I,i}^r = \sigma, \qquad \text{tất cả các tế bào} \tag{17.40}$$

Một khi xác định được giá trị trung bình và độ lệch chuẩn của tín hiệu nhiễu đồng kênh, thì dùng phương pháp Wilkinsson hoặc Schwartz và Yeh để tính giá trị trung bình m^f_I và m^r_I, và độ lệch chuẩn σ^f_I và σ^r_I theo dB của nhiễu đồng kênh tổng trên cả 2 tuyến.

Tính các mô men của các tín hiệu trên *tuyến đi* bằng đoạn mã Matlab sau:

```
% --- NGƯỜI DÙNG MONG MUỐN --------
m_S_fwd = P_BS - 10*K*n_path*log10(des_user_r);
% ----- Các NGƯỜI DÙNG ĐỒNG KÊNH ----
% ----- Vị trí của MS mong muốn theo các ô đồng kênh---
```

```
aux_01 = ((des_user_position(1)-bs_position(:,1)+ sqrt(-
1)*des_user_position(2)-bs_position(:,2)));
beta_fwd = angle(aux_01);
d_I_fwd = abs(aux_01);


% --- Tính độ lợi annten tại các ô đồng kênh
clear gain_fwd;
for k = 1:n_co_ch_users
 if (beta_fwd(k)> = sector_min(sector,sec)) & (beta_fwd(k)<
sector_max(sector,sec))
 gain_fwd(k) = in_beam;
 else
 gain_fwd(k) = out_beam;
 end
end
% ----- Tính các giá trị trung bình và độ lệch chuẩn---
m_I_fwd = P_BS - 10*K*n_path*log10(d_I_fwd) + gain_fwd.';
sigma_I_fwd = sigma_int*ones(length(m_I_fwd),1);
[ m_I_total_fwd,sigma_I_total_fwd] =
wilkinson(m_I_fwd,sigma_I_fwd,corr_fwd);
```

Tương tự, tính các mô men của các tín hiệu trên *tuyến về* bằng đoạn mã Matlab sau:

```
% --- NGƯỜI DÙNG MONG MUỐN ----
m_S_rev = P_MS - 10*K*n_path*log10(des_user_r);
% ----- CÁC NGƯỜI DÙNG ĐỒNG KÊNH ---
% ------ Vị trí của các người dùng đồng kênh ----
aux_02 = (co_ch_user_position(:,1)+ sqrt(-1)*co_ch_user_position(:,2));
beta_rev = angle(aux_02);
d_I_rev = abs(aux_02);
% -----Tính độ lợi ăng ten tai ô trung tâm ---
clear gain_rev;
for k = 1:n_co_ch_users
 if (beta_rev(k)> = sector_min(sector,sec)) & (beta_rev(k)<
sector_max(sector,sec))
 gain_rev(k) = in_beam;
 else
 gain_rev(k) = out_beam;
```

```
 end
end
% ----- Tính các giá trị trung bình và độ lệch chuẩn ---
m_I_rev = P_MS - 10*K*n_path*log10(d_I_rev) + gain_rev.';
sigma_I_rev = sigma_int*ones(length(m_I_rev),1);
[m_I_total_rev,sigma_I_total_rev] =
wilkinson(m_I_rev,sigma_I_rev,corr_rev);
```

Lưu ý rằng, trong ví dụ này ta dùng phương pháp Wilkinson để tính giá trị trung bình và độ lệch chuẩn của nhiễu tổng. Mã Matlab cho phương pháp Wilkinson được cho bởi file **NVD17_ Wilkinson .m** có trong Phụ lục 17A.

• ***Các bước tính toán giá trị thống kê của SIR***

Bước 1: *Tính giá trị trung bình và độ lệch chuẩn của SIR*:

Trong giải pháp này ta tính các giá trị trung bình m^f_{SIR} và m^r_{SIR} và các độ lệch chuẩn σ_{SIR} và σ'_{SIR} của *SIR* theo dB trên cả 2 tuyến đi và về. Các mô men này sẽ được dùng để tính xác suất tin cậy và ngừng hoạt động. Vì tín hiệu mong muốn và tín hiệu nhiễu đồng kênh tổng là các biến ngẫu nhiên chuẩn Gausơ theo dB, nên *SIR* theo dB cũng là một biến ngẫu nhiên chuẩn Gausơ. SIR^f trên hướng đi được cho bởi:

$$SIR^f = S^f_{dB} - I^f_{dB} \tag{17.41}$$

Với giá trị trung bình và độ lệch chuẩn đơn vị dB được cho bởi:

$$m^f_{SIR} = m^f_S - m^f_I \tag{17.42}$$

$$\sigma^f_{SIR} = \sqrt{(\sigma^f_S)^2 + (\sigma^f_I)^2} \tag{17.43}$$

Các phương trình (17.41),(17.42),(17.43) cũng được dùng cho tuyến về. Các kết quả của bước này là m^f_{SIR}, m^r_{SIR}, σ_{SIR} *và* σ'_{SIR}. Mã chương trình Matlab để mô phỏng phần này như sau:

```
m_SIR_fwd(1) = m_S_fwd - m_I_total_fwd;
sigma_SIR_fwd(1) = sqrt(sigma_S^2 + sigma_I_total_fwd^2 -
2*corr_fwd*sigma_S*sigma_I_total_fwd);
m_SIR_rev(1) = m_S_rev - m_I_total_rev;
sigma_SIR_rev(1) = sqrt(sigma_S^2 + sigma_I_total_rev^2 -
2*corr_rev*sigma_S*sigma_I_total_rev);
```

Lưu ý rằng, tại điểm này trong mô phỏng ta bắt đầu tập hợp các kết quả mô phỏng. Trong bước 1 ta lưu trữ các giá trị trung bình và các độ lệch chuẩn trong các mảng được kiểm soát bởi chỉ số *i*.

Bước 2

Phương pháp A: *Lấy mẫu SIR dùng trung bình và độ lệch chuẩn của SIR*

Vì đã biết các trung bình m^f_{SIR} và m^r_{SIR} và các độ lệch chuẩn σ_{SIR} và σ_{SIR} của *SIR* trên cả 2 tuyến từ bước 1, *nên* có thể lấy mẫu các quá trình ngẫu nhiên chuẩn Gausơ SIR^f và SIR^r. Do đó

mỗi chớp ảnh sẽ được liên kết với các mẫu của SIR trên mỗi hướng (ký hiệu là SIR^f_{2A} và SIR^r_{2A}). Các mẫu này là kết quả lấy mẫu của một quá trình ngẫu nhiên chuẩn Gausơ có trung bình m_{SIR} và độ lệch chuẩn σ_{SIR}. Kết quả của bước 2-phương pháp A là các giá trị cho SIR^f_{2A} và SIR^r_{2A} như được xác định bởi mã chương trình Matlab dưới đây. Để được tường minh về quá trình tạo các biến ngẫu nhiên phân bố chuẩn Gausơ này, ta nên nghiên phần help của nó có trong Matlab.

```
% Tạo các biến ngẫu nhiên SIRf2A và SIRr2A phân bố chuẩn Gausơ có các giá
% trị trung bình và độ lệch chuẩn
SIR_fwd_2(1) = normrnd(m_SIR_fwd(1),sigma_SIR_fwd(1));
SIR_rev_2(1) = normrnd(m_SIR_rev(1),sigma_SIR_rev(1))
```

Phương pháp B: *Lấy mẫu SIR dùng trung bình và độ lệch chuẩn của các tín hiệu nhiễu riêng lẻ và tín hiệu mong muốn:*

Tại điểm này, ta đã coi rằng nhiễu đồng kênh tổng được mô hình hóa là tổng các tín hiệu nhiễu đồng kênh riêng lẻ được phân bố loganormal hoặc phân bố chuẩn Gausơ được biểu diễn theo dB. Trong bước 2, phương pháp B, *ta không thực hiện bất kỳ giả định nào đối với phân bố của nhiễu tổng*. Mẫu của mỗi tín hiệu thu tại MS (hướng đi) hoặc tại BS (hướng về) ở tế bào trung tâm được xác định, giả sử các tín hiệu tính theo dB được phân bố chuẩn Gausơ có các mô men S^f, S^r, I^f_i và I^r_i. Theo đó, các tín hiệu nhiễu tổng trên mỗi tuyến được tính như sau:

$$I^f = 10\log_{10}\left(\sum_{i=1}^{6} 10^{\frac{I_i^f}{10}}\right) \quad (17.44)$$

$$I^r = 10\log_{10}\left(\sum_{i=1}^{6} 10^{\frac{I_i^r}{10}}\right) \quad (17.45)$$

Các mẫu của *SIR* trên cả 2 tuyến (ký hiệu là SIR^f_{2B} *và* SIR^r_{2B}) được tính bởi:

$$SIR^f_{2B} = S^f - I^f \quad (17.46)$$

$$SIR^r_{2B} = S^r - I^r \quad (17.47)$$

Mã chương trình Matlab như sau:

```
des_sig_spl_fwd = normrnd(m_S_fwd,sigma_S);
int_sig_spl_fwd = normrnd(m_I_fwd,sigma_I_fwd);
tot_int_sig_spl_fwd = 10*log10(sum(10.^(int_sig_spl_fwd/10)));
SIR_spl_fwd_2B(1) = des_sig_spl_fwd - tot_int_sig_spl_fwd;
des_sig_spl_rev = normrnd(m_S_rev,sigma_S);
int_sig_spl_rev = normrnd(m_I_rev,sigma_I_rev);
tot_int_sig_spl_rev = 10*log10(sum(10.^(int_sig_spl_rev/10)));
SIR_spl_rev_2B(1) = des_sig_spl_rev - tot_int_sig_spl_rev;
```

Tại đây, ta đã hoàn thành việc xử lý *một chớp ảnh*. Cần phải thực hiện *M-1* chớp ảnh còn lại để hoàn tất toàn bộ mô phỏng để tạo ra tập hợp có độ dài *M* của

Từ bước 1: Các giá trị trung bình m^{f}_{SIR} và m^{r}_{SIR} và các độ lệch chuẩn σ^{f}_{SIR} và σ^{r}_{SIR} của *SIR* trên mỗi tuyến

Từ bước 2, phương pháp A: Các mẫu của SIR trên cả 2 tuyến, SIR^{f}_{2A} và SIR^{r}_{2A}

Từ bước 2, phương pháp B: Các mẫu của SIR trên cả 2 tuyến, SIR^{f}_{2B} và SIR^{r}_{2B}

17.3.2. Xử lý các kết quả mô phỏng

Bằng cách xử lý các kết quả mô phỏng, ta có thể ước tính hiệu năng toàn hệ thống tế bào ở dạng xác suất ngừng hoạt động và các bộ chỉ thị hiệu năng khác. Trong phần này sẽ trình bày một số kết quả mô phỏng để so sánh hiệu năng hệ thống tế bào hoạt động theo 6 cấu hình khác dưới đây:

- *Kích cỡ cụm N = 4 và các ăng ten trạm gốc BS đẳng hướng*
- *Kích cỡ cụm N = 4 và phân đoạn 120°*
- *Kích cỡ cụm N = 4 và phân đoạn 60°*
- *Kích cỡ cụm N = 7 và các ăng ten trạm gốc BS đẳng hướng*
- *Kích cỡ cụm N = 7 và phân đoạn 120°*
- *Kích cỡ cụm N = 7 và phân đoạn 60°*

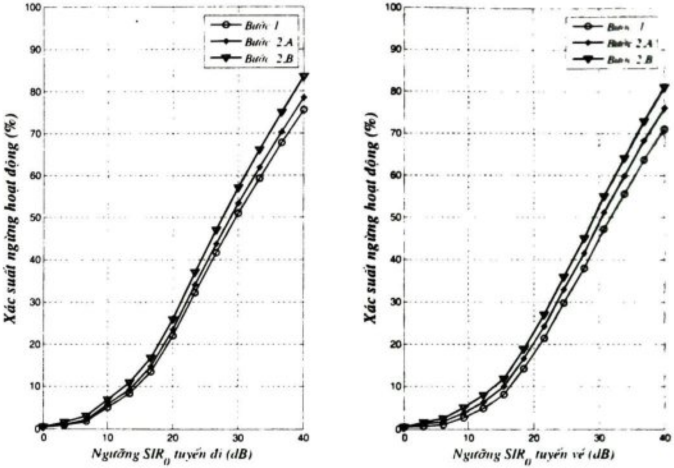
Các ăng ten phân đoạn tại các BS có tỉ số trước sau 30 dB. Độ lệch chuẩn che chắn được đặt là 8 dB và số mũ suy hao truyền sóng $\gamma = 4$. Để đạt được các kết quả thống kê hợp lý cần mô phỏng 1000 chớp ảnh.

Xác suất ngừng hoạt động

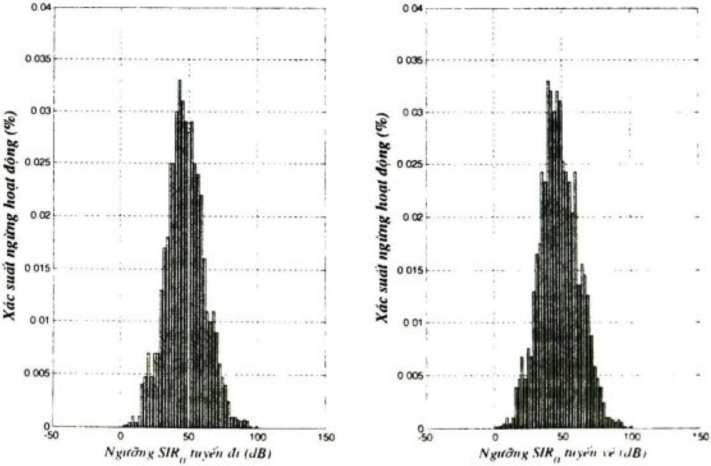
Giá trị trung bình và độ lệch chuẩn của *SIR* biểu diễn theo đơn vị dB đã được tính cho mỗi chớp ảnh trong mô phỏng ở bước 1. Các mô men này tương ứng với *SIR* đo được ở MS và BS tại tế bào trung tâm cho một chớp ảnh vị trí cụ thể của các MS đồng kênh. Vì vậy, ta có thể tính xác suất ngừng hoạt động của hệ thống tế bào tại MS và BS cho một tình huống cho trước. Xác suất ngừng hoạt động đã được định nghĩa ở phần trước, để tiện được nhắc lại:

$$\begin{aligned} P_{\text{ngừng hoạt động}}(SIR_0) &= \Pr\left[SIR < SIR_0\right] = \int_0^{SIR_0} \frac{1}{\sqrt{2\pi}\sigma_{SIR}} e^{\frac{-(x-m_{SIR})^2}{2\sigma_{SIR}^2}} dx \\ &= 1 - Q\left(\frac{SIR_0 - m_{SIR}}{\sigma_{SIR}}\right) \end{aligned} \tag{17.48}$$

Q(.) là hàm *Q* Gausơ. Do đó sử dụng giá trị trung bình và độ lệch chuẩn được tính tại mỗi chớp ảnh từ bước 1, ta nhận được *một mẫu* của xác suất ngừng hoạt động. Mô phỏng xác suất ngừng hoạt động trung bình của hệ thống tế bào $\overline{P}_{\text{ngừng hoạt động}}(SIR_0)$, được tính bằng cách lấy trung bình các mẫu của các xác suất ngừng hoạt động $P_{\text{ngừng hoạt động}}(SIR_0)$ mã đã được tính toán cho mỗi chớp ảnh. Các xác suất ngừng hoạt động trung bình thường được coi là xác suất ngừng hoạt động trung bình *vùng* vì mỗi phần tử trong quá trình lấy trung bình tương ứng với một vị trí trong diện tích tế bào. Các xác suất ngừng hoạt động trung bình tại các ngưỡng khác nhau được tính bằng cách dùng các kết quả từ bước 1 đối với kích cỡ cụm $N = 7$ và phân đoạn 120^0 được cho ở hình 17.13. Hình 17.13 cũng cho thấy các xác suất ngừng hoạt động trung bình đối với kích cỡ cụm $N = 7$ và phân đoạn 120^0 được tính từ các kết quả bước 2, phương pháp A và B.



Hình 17.13: Xác suất ngừng hoạt động trung bình trên cả hai tuyến đi và về dùng các phương pháp đã được mô tả (kích cỡ cụm N = 7, phân đoạn 120^0, $\sigma = 8$dB và $\gamma = 4$)

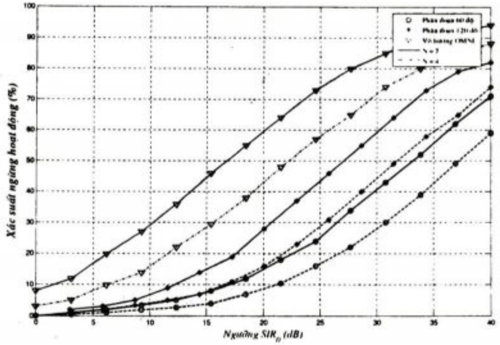


Hình 17.14 : Hoành đồ của các mẫu SIR từ giải pháp 2 cho hai tuyến đi và về dùng bước 2 phương pháp A (kích cỡ cụm N = 7, phân đoạn 120^0, $\sigma = 8$dB và $\gamma = 4$)

Xác suất ngừng hoạt động trung bình cũng có thể được tính từ các kết quả bước 2 phương pháp A hoặc B. Nó cung cấp tập hợp các mẫu của *SIR* trên cả hai tuyến. Trên cơ sở các tập hợp

này, ta có thể ước tính hàm mật độ xác suất của *SIR trung bình vùng*. Muốn vậy, trước hết phải tính hoành đồ của các tập hợp. Ví dụ hình 17.14 cho thấy các hoành đồ của các mẫu của SIR trung bình vùng có được từ bước 2, phương pháp A. Các hoành đồ này xấp xỉ hàm pdf đúng của SIR trung bình vùng. Sử dụng hoành đồ ta có thể ước tính các xác suất ngừng hoạt động trung bình trên cả 2 tuyến.

Tại đây ta có thể so sánh hiệu năng 6 cấu hình được mô phỏng. Hình 17.15 cho thấy xác suất ngừng hoạt động trung bình của mỗi cấu hình, sử dụng bước 1 trên hướng đi. Như được mong đợi, việc phân đoạn cải thiện hiệu năng hệ thống tế bào, tức là giảm được xác suất *SIR* dưới một ngưỡng cho trước. Chẳng hạn, hệ thống sử dụng kích cỡ cụm $N = 7$ và 3 sector trên một tế bào tốt hơn (về chất lượng tuyến) so với hệ thống sử dụng cụm cỡ $N = 7$ nhưng với các ăng ten đẳng hướng. Xác suất ngừng hoạt động tại $SIR_0 = 18$dB (ngưỡng thường được dùng trong AMPS) đối với hệ thống thứ nhất là 15% trong khi đó hệ thống thứ hai là 35%.



Hình 17.15 : Xác suất ngừng hoạt động trên tuyến đi cho các cấu hình

Tuy nhiên, phân đoạn làm giảm dung lượng hệ thống ở dạng lưu lượng mang tối đa. Để minh họa, bảng 17.2 cho thấy lưu lượng mang trên một tế bào đối với mỗi cấu hình ứng với xác suất ngừng hoạt động tại $SIR_0 = 18$ dB nhận được từ hình 17.15. Lưu lượng mang được tính toán theo giả định: có 395 kênh khả dụng cho toàn bộ hệ thống và xác suất nghẽn chấp nhận được là 0,02. Thấy rõ, sự dung hòa giữa dung lượng và chất lượng tuyến. Khi cải thiện được dung lượng thì chất lượng đường truyền giảm xuống.

Hiệu năng hệ thống trên diện tích tế bào

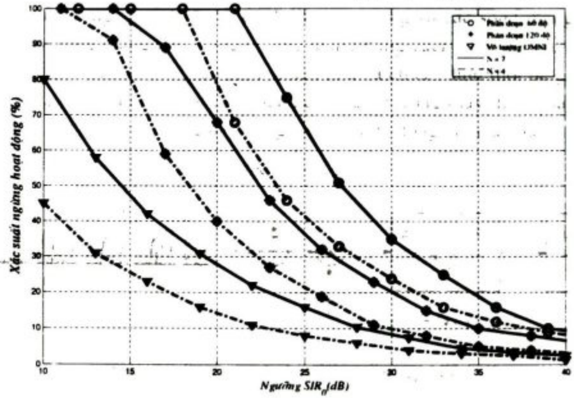
Xác suất ngừng hoạt động trung bình, như đã đề cập và tính toán ở trên, cho biết hiệu năng hệ thống tế bào được lấy trung bình trên diện tích tế bào. Kết quả là, tồn tại một xác suất ngừng hoạt động cao (không mong muốn) tại một vị trí cho trước có thể được bù bởi một xác suất ngừng hoạt động thấp tại một vị trí khác. Đôi khi, khi phân tích hiệu năng ta cần phải đánh giá phần trăm diện tích tế bào ở đó hiệu năng hệ thống (xác suất ngừng hoạt động) có thể chấp nhận được hay không. Sự đánh giá này có thể được tính toán từ các kết quả mô phỏng.

Xác suất tin cậy $P_{tin\ cậy}(SIR_0)$ được định nghĩa là xác suất mà *SIR* lớn hơn một ngưỡng SIR_0 cho trước. Theo đó, tại một vị trí cho trước ta có:

$$P_{\text{tin cậy}}(SIR_0) = \Pr\left[SIR > SIR_0\right] = Q\left(\frac{SIR_0 - m_{SIR}}{\sigma_{SIR}}\right) = 1 - P_{\text{ngừng hoạt động}}(SIR_0) \quad (17.49)$$

Bảng 17.2: Lưu lượng mang trên tế bào và xác suất ngừng hoạt động

Ăng ten	Kích cỡ cụm N = 4	Kích cỡ cụm N = 7
Vô hướng	86,0 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,52	45,9 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,35
3-sector	71,2 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,22	24,5 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,15
6-sector	59,0 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,13	26,1 Erg @ $P_{ngừng\ hoạt\ động}$ = 0,07



Hình 17.16: Tỉ lệ phần trăm của diện tích tế bào ở đó $P_{rela}(SIR_0) > 0,75$ được duy trì trên tuyến đi cho tất cả các cấu hình được mô phỏng ($\sigma = 8$ dB và $\gamma = 4$)

Tại đây, ta coi hiệu năng của hệ thống là chấp nhận được nếu xác suất tin cậy $P_{tin\ cậy}(SIR_0)$ tại ngưỡng SIR_0 cho trước lớn hơn ngưỡng P_{min}. Sau đó sử dụng các giá trị $P_{tin\ cậy}(SIR_0)$ đã tính được để ước tính phần trăm diện tích tế bào ở đó có hiệu năng hệ thống chấp nhận được (tức là nơi có $P_{tin\ cậy}(SIR_0) > P_{min}$) bằng cách tính:

$$P_{\text{diện tích}}(SIR_0) = \frac{\text{Số vị trí có } P_{\text{tin cậy}}(SIR_0) > P_{\min}}{\text{Tổng số vị trí}} \quad (17.50)$$

Lưu ý rằng, việc tính toán $P_{diện\ tích}(SIR_0)$ chỉ có thể thực hiện được khi dùng các kết quả của bước 1 trong mô phỏng vì cần có giá trị trung bình và độ lệch chuẩn của SIR. Hình 17.16 cho thấy tỉ lệ phần trăm của diện tích tế bào có $P_{tin\ cậy}(SIR_0) > 0,75$ trên tuyến đi cho tất cả các cấu hình đã được mô phỏng.

Chương 18
MÔ PHỎNG HỆ THỐNG CDMA VÀ FDM ĐƠN GIẢN

18.1. Mở đầu

Kết thúc phương pháp luận và kỹ thuật mô phỏng là hai mô phỏng điển hình:

Mô phỏng hệ thống CDMA hoạt động trong môi trường kênh pha đing đa đường: Nhiễu và tạp âm nhiệt cũng được xét đến trong mô hình mô phỏng. Mô hình mô phỏng cho phép khảo sát hiệu năng BER ở dạng: ***(i)*** BER là một hàm của mức tạp âm nhiệt (khảo sát P_E theo E_b/N_0); ***(ii)*** BER là một hàm của hệ số trải phổ (độ lợi xử lý trải phổ); ***(iii)*** BER là một hàm của nhiễu. Mặc dù hệ thống được trình bày ở đây phức tạp hơn các chương trước nhưng vẫn được đơn giản hóa để chạy chương trình Matlab với thời gian chấp nhận được. Mô phỏng Monte Carlo được dùng cho trường hợp này.

Mô phỏng hệ thống truyền thông vệ tinh đa kênh FDM: Giả sử có một bộ khuếch đại công suất phi tuyến, chương trình mô phỏng đánh giá ảnh hưởng tính phi tuyến lên hiệu năng hệ thống bằng kỹ thuật mô phỏng bán phân tích. Do mô phỏng bán phân tích nhanh hơn mô phỏng Monte Carlo, nên việc đơn giản hóa mô hình mô phỏng là không quan trọng như trong trường hợp CDMA.

18.2. Hệ thống đa truy nhập phân chia theo mã CDMA

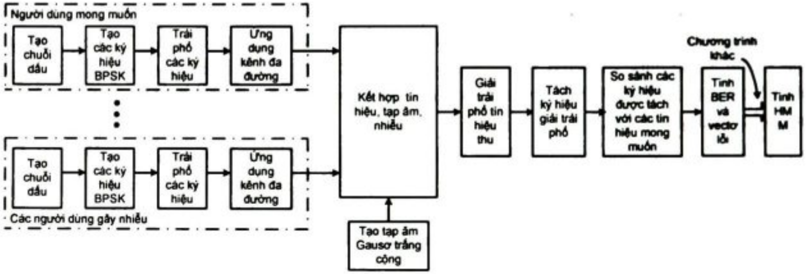
Bộ mô phỏng đa truy nhập phân chia theo mã CDMA ***phỏng tạo*** tuyến truyền thông CDMA đơn giản. Các hệ thống CDMA được triển khai ở đây cũng được dùng để xây dựng các khối chức năng cho các hệ thống tiên tiến hơn. Sơ đồ khối mô hình mô phỏng được minh họa ở hình 8.1. Các ảnh hưởng của kênh AWGN, kênh đa đường và nhiễu đa truy nhập MAI cũng được xét trong bộ mô phỏng này. Theo đó, cho phép ước tính BER là một hàm của: ***(i)*** tỷ số năng lượng bit trên mật độ phổ tạp âm E_b/N_0; ***(ii)*** số nguồn nhiễu *NoI*; ***(iii)*** hệ số trải phổ SF. Nó có thể áp dụng cho bất cứ kênh đa đường nào miễn là không nhiều hơn năm thành phần đa đường.

18.2.1. Hệ thống

Ở đa truy cập phân chia theo tần số FDMA hoặc đa truy cập phân chia theo thời gian TDMA, việc phân tách người dùng vô tuyến bằng cách gán các băng tần số khác nhau hoặc các khe thời gian khác nhau, còn CDMA thực hiện phân tách người dùng bằng cách gán các mã khác nhau. Theo đó, một hệ thống CDMA sẽ có toàn quyền sử dụng một tập mã gồm tập hợp các chuỗi giả ngẫu nhiên. Tập hợp các chuỗi này phải có các tính tương quan chéo thấp để máy thu có thể chấp nhận tín hiệu từ người dùng mong muốn trong khi đó từ chối tín hiệu từ người dùng không mong muốn (người dùng đồng kênh). Hơn nữa, mỗi chuỗi trong tập hợp này phải có các thuộc tính tương quan tốt để cho phép tìm được các ảnh hưởng đa đường trong các kênh vô tuyến mà sẽ được khai thác bởi máy thu RAKE hoặc các kỹ thuật thu không gian-thời gian.

Hệ thống CDMA gán mỗi người dùng một chuỗi mã riêng từ tập chuỗi mã của hệ thống. Chuỗi mã này được sử dụng để tạo dạng sóng trải phổ, tốc độ ký hiệu (tốc độ chip) của chúng cao hơn nhiều tốc độ ký hiệu của tín hiệu mang tin (tốc độ ký hiệu). Tỷ lệ tốc độ chip trên tốc độ ký hiệu mang tin được gọi là hệ số trải phổ hay độ lợi xử lý. Các hệ số trải phổ có giá trị điển hình từ 8 đến 512.

Các hệ thống CDMA điển hình sử dụng các kỹ thuật trải phổ chuỗi trực tiếp. Dạng sóng mang tin được điều chế (được nhân) với dạng sóng trải phổ dành cho (được cấp) người dùng tạo ra tín hiệu trải phổ, sau đó được phát xạ vào kênh truyền. Tại đầu ra kênh, máy thu (giả sử máy tương quan) lấy tương quan tín hiệu thu với dạng sóng trải phổ của người dùng. Theo đó, cho phép tách tín hiệu mang tin của người dùng mong muốn bằng bộ tương quan, đồng thời loại bỏ tất cả tín hiệu người dùng khác (nhiễu).



Hình 18.1: Phương pháp luận mô phỏng cho hệ thống CDMA

Để tạo mô phỏng tuyến truyền thông CDMA có hiệu quả tính toán cao, ta cần phải kết hợp chặt chẽ các đơn giản hóa/giả định sau:

1. Coi điều khiển công suất là lý tưởng và mỗi tín hiệu đến máy thu có công suất trung bình bằng nhau.

2. Chuỗi dấu/trải phổ cho tín hiệu mong muốn là chuỗi PN như được đề cập trong chương 7. Theo đó, các hệ số trải phổ được phép phải bằng $2^R - 1$, trong đó R là một số nguyên dương.

3. Chuỗi trải phổ/dấu được sử dụng cho mỗi tín hiệu MAI cũng là một chuỗi PN. Tuy nhiên, nó được dịch vòng sao cho đạt được thuộc tính tương quan mong muốn. Đặc biệt lưu ý rằng, mặc dù đảm bảo những tính tương quan mong muốn trong môi trường kênh AWGN nhưng có thể mất tính tương quan trong môi trường kênh đa đường như được mô hình hóa ở đây (vì các tín hiệu nhiễu và mong muốn được phân tách theo thời gian bằng một số nguyên lần chip). Cũng có thể xảy ra trường hợp các trễ này (dịch chuỗi mã) bằng với các trễ đa đường, dẫn đến các kết quả không khả quan khi mô phỏng đồng thời nhiễu và đa đường.

4. Dùng điều chế khoá dịch pha nhị phân BPSK và bỏ qua định dạng xung.

5. Kênh đa đường biểu lộ pha đing Rayleigh trên mỗi thành phần đa đường ngoại trừ thành phần đi thẳng LOS. Các tín hiệu nhiễu không biểu lộ pha đing.

6. Trễ của mỗi thành phần đa đường chỉ được xét trong bội số nguyên lần khoảng thời gian chip.

7. Tín hiệu mong muốn và tất cả các tín hiệu nhiễu đa truy nhập MAI được đồng bộ theo chip tại máy thu.

8. Dùng máy thu tương quan. Không dùng máy thu RAKE.

Bỏ qua tính chi tiết về việc mô hình hóa định dạng xung và các bộ lọc thích hợp tại máy thu, chúng có ảnh hưởng lớn lên tốc độ mô phỏng (vì các dạng sóng cơ bản có thể được thực hiện một mẫu trên một chip). Có được các kết quả hợp lý nếu thực hiện định dạng xung ISI = 0 trong hệ thống.

Thực tế, thường dùng các kết quả đo kênh để xác định lý lịch trễ công suất và thực hiện mô phỏng bằng cách sử dụng phép xấp xỉ phù hợp với lý lịch trễ công suất đo. Điều này đảm bảo mô hình mô phỏng sẽ đặc tính hóa chính xác môi trường hệ thống vô tuyến được triển khai. Tuy nhiên, khi này cần lưu ý rằng, phải xác định rõ lý lịch trễ và lý lịch công suất sao cho thỏa mãn hệ số Rice K. Muốn vậy, phải hiểu quan hệ giữa hiệu năng của hệ thống và hệ số Rice K. Chẳng hạn, khi K lớn thì hiệu năng hệ thống về cơ bản tương đương với hiệu năng trong môi trường kênh AWGN. Khi $K \cong 0$, môi trường truyền sóng là pha đing Rayleigh và có thể là pha đing phẳng hoặc pha đing chọn lọc tần số tùy vào số thành phần đa đường và trễ tương ứng với các thành phần đa đường này.

Trong mô phỏng này, lý lịch trễ đa đường chỉ là một véc-tơ các số nguyên trong đó mỗi số nguyên thể hiện cho trễ của mỗi thành phần đa đường ở dạng các chu kỳ chip (một chu kỳ chip bằng một chu kỳ ký hiệu chia cho hệ số trải phổ). Trong mô phỏng, ta coi phần tử thứ nhất trong véc tơ này tương ứng với thành đi thẳng LOS, và các phần tử còn lại là các thành phần tán xạ (đa đường). Độ dài (kích thước) của véc tơ này xác định tổng số các thành phần thu được (gồm thành phần LOS và các thành phần tán xạ).

Một khi định rõ lý lịch trễ và hệ số Rice K, sẽ tính được lý lịch công suất ứng với hệ số K đó. Lý lịch công suất thu cùng với lý lịch trễ có thể được hiển thị bởi chương trình hậu xử lý vì vậy cho phép trực quan hóa chúng cũng như các ảnh hưởng của chúng lên tín hiệu tại các điểm khác nhau của mô hình mô phỏng.

Tại đây ta nghiên cứu cách xác định lý lịch công suất sao cho thỏa mãn hệ số K. Muốn vậy,

Trước hết là, tạo biên độ trung bình của mỗi thành phần tán xạ bằng bộ tạo số ngẫu nhiên (có các đầu ra được phân bố đều trong khoảng [0,1]). Đặt:

$$A_{\text{tán xạ}} = [\alpha_1 \quad \alpha_2 \quad \dots \quad \alpha_{M-1}] \tag{18.1}$$

Biểu thị biên độ trung bình của mỗi thành phần tán xạ. Đặc biệt lưu ý rằng M là tổng số các thành phần đa đường (gồm M-1 tín hiệu tán xạ và một thành phần LOS).

Sau đó là, tính năng lượng tán xạ trung bình theo:

$$E_{\text{tán xạ}} = \sum_{i=1}^{M-1} \alpha_i^2 \tag{18.2}$$

Năng lượng trung bình trong thành phần LOS được tính như sau:

$$K = \frac{\text{Công suất trong đường dẫn trội}}{\text{Công suất trong các đường tán xạ}} = \frac{A^2}{2\sigma^2} = \frac{E_{LOS}}{E_{\text{tán xạ}}} \tag{18.3}$$

$$\Rightarrow E_{LOS} = K \times E_{\text{tán xạ}}$$

Trong đó K là hệ số Ricean. Biên độ trung bình của thành phần LOS là:

$$\alpha_{LOS} = \sqrt{E_{LOS}} \tag{18.4}$$

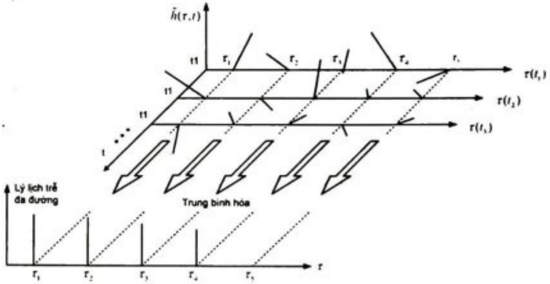
Lý lịch biên độ trung bình đạt được bằng cách thêm a_{LOS} vào véc tơ các thành phần tán xạ A_{scat}.

Cuối cùng là, chuẩn hóa véc tơ này sao cho thành phần đa đường mạnh nhất có giá trị 1,0 nghĩa là:

$$A = \frac{[\alpha_{LOS} \quad \alpha_1 \quad \alpha_2 \quad \ldots \quad \alpha_{M-1}]}{\max\{[\alpha_{LOS} \quad \alpha_1 \quad \alpha_2 \quad \ldots\alpha_{M-1}]\}} = \left[\underbrace{a_{LOS}} \quad a_1 \quad a_2 \quad \ldots \quad a_{M-1}\right] \tag{18.5}$$

Theo đó tìm được lý lịch công suất bằng cách lấy bình phương các phần tử trong A nghĩa là:

$$P = \left[a_{LOS}^2 \quad a_1^2 \quad a_2^2 \quad \ldots \quad a_{M-1}^2\right] \tag{18.6}$$



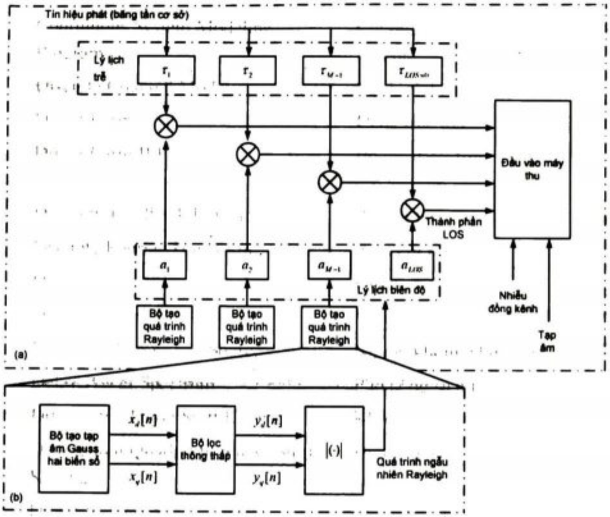
Hình 18.2: Minh họa đáp ứng xung kim kênh và lấy trung bình các thành phần tán xạ

Hình 18.2 minh họa đáp ứng xung kim, các thành phần tán xạ và việc lấy trung bình các thành phần tán xạ.

Ta thấy rõ: ***(i)*** khi $K > 1$, thành phần LOS sẽ là thành phần mạnh nhất $a_{LOS} = 1,0$; ***(ii)*** khi $K \leq 1$ thì tình huống không rõ ràng. Khi này, năng lượng tán xạ trung bình vượt quá năng lượng trung bình trong thành phần LOS. Tuy nhiên, năng lượng tán xạ trung bình $E_{\text{tán xạ}}$ có thể trải rộng vài thành phần đa đường. Cũng vậy, thành phần LOS có thể là hoặc không là thành phần đa đường mạnh nhất. Tất nhiên, lý lịch công suất cho phép xác định thành phần mạnh nhất.

Một khi định rõ lý lịch trễ và lý lịch biên độ (công suất), thì ta xác định được mô hình kênh. Mô hình kênh được minh hoạ trong hình 18.3 (a). Bộ tạo quá trình ngẫu nhiên Rayleigh được minh hoạ trong hình 18.3 (b), việc thực thi nó được được mô tả trong chương 7. Theo đó,

hai biến ngẫu nhiên Gausơ độc lập $x_d[n]$ và $x_q[n]$ được tạo ra và được lọc để cho ra $y_d[n]$ và $y_q[n]$. Loại bộ lọc được dùng trong bộ tạo quá trình Rayleigh là tùy ý, được chọn là một bộ lọc Chebyshev bậc bốn có độ gợn băng thông là 0,5 dB và độ rộng băng bằng với tần số Doppler. Độ lớn $\sqrt{y_d^2[n]+y_q^2[n]}$ là một biến ngẫu nhiên Rayleigh.



Hình 18.3: Minh họa (a) Mô hình kênh CDMA;
(b) Bộ tạo quá trình ngẫu nhiên Rayleigh

Để mô phỏng phải định rõ tham số E_b/N_0. Tham số này xác định tỉ số năng lượng bit trung bình trong thành phần đa đường mạnh nhất (là thành phần LOS khi K>1) trên năng lượng tạp âm nhiệt. Lưu ý rằng, công suất trung bình của thành phần đa đường mạnh nhất sẽ được chuẩn hoá bằng 1. Do đó, mô phỏng lấy tỉ số năng lượng của tạp âm (nhiệt) Gausơ trắng sao cho E_b/N_0 trung bình trong thành phần đa đường mạnh nhất bằng tham số E_b/N_0.

18.2.2. Chương trình mô phỏng

Mô phỏng CDMA có cấu trúc theo từng chuỗi khối, trong đó xử lý lần lượt từng khối 1000 ký hiệu. Như đã đề cập ở chương 10, dùng cấu trúc chuỗi khối cho ta hiệu quả tính toán. Mã chương trình mô phỏng đầy đủ được cho ở file **NVD18_cdmasim** trong Phụ lục 18 A. Khi thực hiện lệnh.

$$\left[\underbrace{\text{BER, Erorrs}}_{\text{Các kết quả mô phỏng}}\right] = \underbrace{\text{NVD18_cdma_sim}}_{\text{Tên hàm}}\left(\underbrace{N, SF, EbN0, NoI, MPathDelay, KFactordB}_{\text{Các tham số đầu vào của hàm}}\right) \quad (18.7)$$

Định nghĩa đầy đủ chương trình mô phỏng chính. Một số chương trình con (hàm con) cùng với các chương trình hậu xử lý (xử lý hiển thị) cũng được cho ở phụ lục 18A. Cũng cần lưu ý rằng bộ mô phỏng được thiết kế theo kiến trúc Modul.

Các tham số đầu vào của chương trình mô phỏng được định nghĩa trong bảng 18.1. Nó rất quan trọng để quan trắc hiệu năng (so sánh hiệu năng theo các tham số đầu vào). Do đó có thể triển khai một chương trình tiền xử lý để xử lý giải giá trị tham số đặc trưng.

Bảng 18.1: Các tham số đặc trưng đầu vào hệ thống CDMA

Tham số	Kiểu	Mô tả	Giới hạn
N	Vô hướng	Số ký hiệu được mô phỏng	Không
SF	Vô hướng	Hệ số trải phổ	$SF = 2^n, \quad n \le 12$
E_b/N_0	Vô hướng	Tỉ số năng lượng bit trên PSD tạp âm	Không
NoI	Vô hướng	Số nguồn nhiễu	$0 \le NoI \le SF - 1$
MPathDelay	Véc tơ	Lý lịch trễ đa đường	Véc tơ các số nguyên không âm đơn điệu tăng
KfactordB	Vô hướng	Hệ số Rice K đơn vị dB	Không

18.2.3. Các mô phỏng

a) Xác nhận tính hợp lệ ranh giới

Để đảm bảo mô phỏng được hiệu chuẩn chính xác, trước tiên bộ mô phỏng CDMA được chạy với hệ số K = 100 dB. Hệ số K này đủ lớn để đảm bảo rằng, dù có xảy ra mức đa đường rất nhỏ đi chăng nữa thì công suất tán xạ bị suy hao đủ lớn để thực hiện lấy xấp xỉ tất cả công suất thu là trong thành phần LOS. Vì vậy, ảnh hưởng làm suy thoái hiệu năng chỉ là tạp âm Gausơ và kết quả xác xuất lỗi của hệ thống là:

$$P_E = Q\left(\sqrt{\frac{2E_b}{N_O}}\right) \tag{18.8}$$

Tại mỗi giá trị của E_b/N_0, chương trình mô phỏng xử lý cho N ký hiệu. Chẳng hạn muốn khảo sát số lỗi (hoặc BER) theo các giá trị E_b/N_0 thay đổi 0 đến 8 dB với mỗi bước 1 dB, tương ứng mỗi giá trị của E_b/N_0 xử lý cho N = 200000 ký hiệu. Chương trình Matlab để chạy mô phỏng hiệu chuẩn là **NVD18_cdmacal.** Lưu ý rằng, trong **NVD18_cdmacal** chứa chương trình **NVD18_cdmasim**

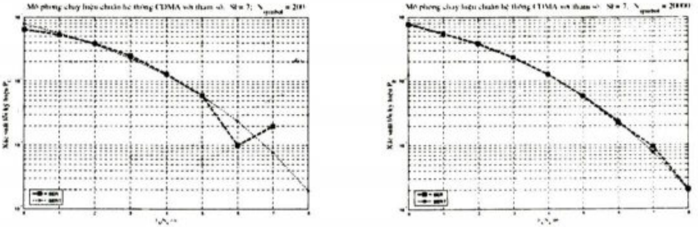
```
>> NVD18_cdmacal
   Nhập số ký hiệu sẽ được xử lý      >      200000
   Nhập véc tơ Eb/N0                  >      0:8
```

Kết quả được cho trong bảng 18.2 và hình 18.4. Số lỗi xuất hiện với mỗi giá trị E_b/N_0 được cho trong bảng 18.2. Dữ liệu này đặc biệt quan trọng việc chạy hiệu chuẩn để đảm bảo số lỗi đủ lớn tại mỗi giá trị E_b/N_0 để có được kết quả thống kê đủ tin cậy. Kết quả mô phỏng khảo sát xác suất lỗi P_E là một hàm của E_b/N_0 được cho trong hình 18.4. Đường nét liền

là kết quả lý thuyết theo công thức (18.8), và các kết quả mô phỏng được chỉ ra bằng hình vuông tại các giá trị nguyên của E_b/N_0. Thấy rõ các kết quả đạt được khá chính xác ngoại trừ $E_b/N_0 = 8$ dB. Bảng 18.2 cho thấy xác xuất lỗi đối với $E_b/N_0 = 8$ được dựa trên 53 lỗi, nó nhỏ so với số các lỗi xuất hiện tại các giá trị E_b/N_0 khác. Mã chương trình Matlab bộ tiền xử lý được cho trong Phụ lục 18A.

Bảng 18.2: Đếm lỗi khi chạy hiệu chuẩn

E_b/N_0 (dB)	0	1	2	3	4	5	6	7	8
Các lỗi	15773	11347	7583	4516	2416	1169	475	159	53



Hình 18.4: Kết mô phỏng chạy hiệu chuẩn

Ta thấy, khi số ký hiệu được mô phỏng lớn (so sánh giữa 200 và 20.000), kết quả mô phỏng chính xác hơn đặc biệt tại các giá trị E_b/N_0 lớn.

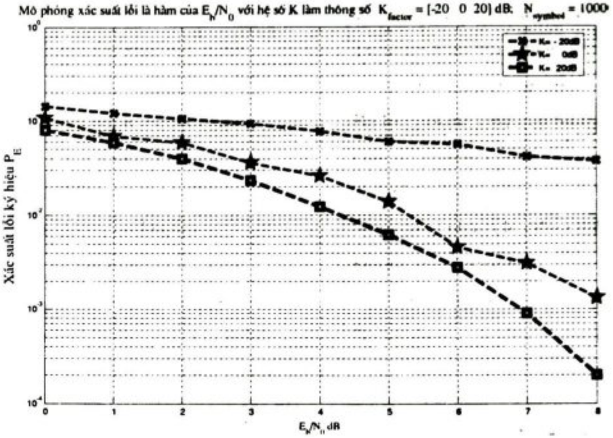
b) Hiệu năng BER là hàm của E_b/N_0 và hệ số Rice K

Mô hình mô phỏng tạo ra BER là một hàm của E_b/N_0 và hệ số K. Theo đó, hai biến được cho trong bảng 18.1 phải được lặp lại cho từng giá trị của K. Cụ thể, với mỗi giá trị của hệ số K, thì E_b/N_0 có giá trị từ 0 dB đến 10 dB với bước thay đổi là 1 dB. Ba giá trị của hệ số K được sử dụng -20 dB, 0dB, 20 dB. Chương trình Matlab bộ xử lý/bộ hậu xử lý **NVD18_cdmaK.m** (trong chương trình này chứa **NVD18_cdmasim**) được dùng để tạo các kết quả này được cho trong Phụ lục 18A. Đoạn hội thoại dưới đây được dùng để nhập các tham số đầu vào chương trình mô phỏng

```
>> NVD18_cdmaK
    Nhập số ký hiệu sẽ được xử lý:        100.000
    Nhập véc tơ Eb/N0:                    0:10
    Nhập véc tơ hệ số K đơn vị dB:        [-20 0 20]
```

Kết quả mô phỏng để nghiên cứu ảnh hưởng của hệ số K lên hiệu năng BER được cho ở hình 18.5 trong đó: ***(i)*** Đường cong trên cùng tương ứng với hệ số K = -20 dB (K = 0,01 lần) cho ta kết quả xấp xỉ kết quả trong môi trường kênh pha đing Rayleigh; ***(ii)*** Đường cong dưới tương ứng với hệ số K = 20 dB (K = 100 lần), cho ta kết quả xấp xỉ hiệu năng trong môi trường

tạp âm Gausơ; *(iii)* Đường cong giữa tương ứng với hệ số $K = 0$ dB ($K = 1$ lần) thường được rút ra khi sử dụng mô phỏng. Các kết quả đối với cả *K* lớn và $K \cong 0$ có thể nhận được từ giải tích và được sử dụng để xác nhận mô phỏng. Lưu ý rằng, đường cong giữa ($K = 1$) không mịn bằng các đường cong khác do bản chất thay đổi theo thời gian của kênh, gần như thể hiện các giá trị tức thời của *K* tại đó công suất trong thành phần LOS xấp xỉ bằng tổng công suất tán xạ. Cần thực hiện mô phỏng rất lâu để làm mịn ảnh hưởng thay đổi theo thời gian này.



Hình 18.5: Xác xuất lỗi là hàm của E_b / N_0 với hệ số Rice K là tham số K = 100 (đường cong dưới), K = 1 (đường cong giữa) và K = 0,01 (đường cong trên cùng))

Có nhiều tiềm năng khác để nghiên cứu. Khi phát triển chương trình tiền xử lý sẽ cho phép nghiên cứu ảnh hưởng của hệ số trải phổ đối với lý lịch trễ tại E_b / N_0 cho trước. Theo đó, ta nên thực hiện mô phỏng tại các giá trị khác nhau của E_b / N_0 ứng với một số lý lịch trễ khác nhau. Nghiên cứu khảo sát theo số lượng các nhiễu.

Mở rộng mô hình mô phỏng CDMA bằng cách đưa thêm các tham số vào mô hình mô phỏng ở dạng các véc tơ của tham số mô phỏng. Theo đó, khi chạy chương trình mô phỏng, chương trình sẽ tạo ra rất nhiều dữ liệu, tất cả đều phải được phân tích trước khi chọn thiết kế "tốt nhất". Kết quả là, nghiên cứu mô phỏng phải được tổ chức cẩn thận sao cho số liệu nhận được từ các mô phỏng khác nhau sẽ được "*khai thác*" để hỗ trợ thiết kế bằng cách phát hiện các tham số tới hạn và các dải giá trị khả chấp nhận cho các tham số này. Việc tổ chức các mô phỏng và khai thác số liệu được tạo ra từ những mô phỏng là ngoài phạm vi của quyển sách này. Tuy nhiên, đây là chủ đề được quan tâm nghiên cứu mô phỏng chi tiết.

18.2.4. Triển khai mô hình Markov

Ta tạo mô hình Markov ẩn HMM cho hệ thống CDMA ở hình 18.1 bằng cách sử dụng các công cụ được đề cập ở chương 15. Để xác định và ước lượng HMM ta triển khai ba chương

trình Matalab sau: ***(i)* Thực hiện** mô phỏng CDMA và xác định mô hình bán Markov (mô hình bán Markov được sử **dụng để** giảm tải tính toán); ***(ii)*** Tạo một véc-tơ lỗi dựa trên ma trận chuyển dịch trạng thái của mô hình bán Markov. Ngoài ra, BER được dựa trên hệ thống CDMA, BER được ***dự đoán*** bởi mô hình bán Markov, được so sánh với BER dựa trên chuỗi lỗi **được** tạo bởi mô hình **bán Markov**; ***(iii)*** So sánh véc-tơ lỗi được tạo bởi mô phỏng ban đầu với véc-tơ lỗi được tạo ra bởi mô hình bán Markov. Cả đại lượng thống kê Pr $\{0^m | 1\}$ và hoành đồ của các độ dài chạy chương trình không lỗi được dùng để trong so sánh. Dưới đây khảo sát các chương trình này.

Chương trình thứ 1: NVD18_cdmahmm1.m

Chương trình này định nghĩa các tham số mô phỏng CDMA và thực hiện mô phỏng. Như được thấy ở (18.7) chương trình **NVD18_cdma** tạo hai đầu ra kết quả từ mô phỏng hệ thống CDMA. Đó là BER và véc-tơ lỗi được dùng để xác định mô hình Markov (được ký hiệu là ErrorRun). Ở đây ta tạo mô hình bán Markov cho kênh hoạt động tại K = 1 (0 dB) và $E_b/N_0 = 5$ dB. Mã chương trình Matlab được cho bởi **NVD18_cdmahmm1.m**.

Chạy chương trình này cho ta mô hình bán Markov dựa trên 50 lần lặp. Kết quả là:

$$\hat{A} = \begin{bmatrix} 0,9494 & 0 & 0,0506 \\ 0 & 0,9714 & 0,0286 \\ 0,2251 & 0,7419 & 0,0330 \end{bmatrix} \qquad (18.9)$$

Lưu ý rằng, rất nhiều biến được lưu vào đĩa. Các biến này được dùng cho hai chương trình tiếp sau (**NVD18_cdmahmm2.m** và **NVD18_cdmahmm3.m**), chúng được lưu sao cho hai chương trình này dễ dàng được gọi ra. Để được tường minh, ta nên nghiên cứu từng lệnh của chương trình và chạy từng lệnh. Sau khi chạy chương trình **NVD18_cdmahmm1.m** kết quả được lưu ở file dữ liệu có tên là **cdmadata1** trong thư mục Work.

Chương trình thứ 2: NVD18_cdmahmm2.m

Trong chương trình thứ hai tạo ra một véc-tơ lỗi bằng cách xử lý chuỗi ký hiệu qua kênh (được biểu diễn bởi mô hình bán Markov và được tạo ra trong chương trình **NVD18_cdmahmm1.m**). Ngoài ra, xác xuất lỗi được tạo ra bằng ba kỹ thuật. *Thứ nhất*, xác suất lỗi bit được tạo ra bởi mô phỏng CDMA ban đầu. *Thứ hai*, xác xuất lỗi bit là BER được dự đoán bởi mô hình *bán* Markov (tăng $\hat{A}$ tới số mũ cao như được mô tả trong chương 15). *Thứ ba*, xác xuất lỗi bit được xác định bằng cách đếm lỗi khi cho một số lượng lớn các ký hiệu (trường hợp này là 25000) qua mô hình kênh bán Markov. Mã chương trình Matlab được cho bởi **NVD18_cdmahmm2.m** trong Phụ lục 18A.

Véc-tơ lỗi dựa trên mô hình Makov ẩn HMM được tạo ra bởi hàm **NVD18_errvector** (là hàm **NVD15_errorvecto** ở chương 15). Chương trình **NVD18_errorvector** được cho trong Phụ lục 18A. Chạy chương trình Matlab sẽ nhận được kết quả:

```
>> NVD18_cdmahmm2
   NN                                        = 25000
   Xác suất lỗi được dự đoán cho hệ thống CDMA:
```

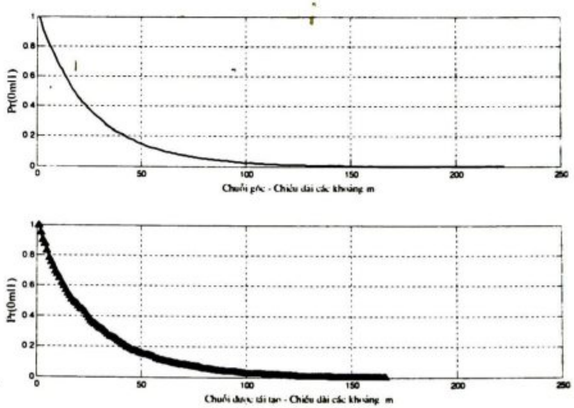
Kết quả từ mô phỏng ban đầu P_E	= 0,03185
Kết quả dự đoán từ mô hình bán Markove P_E	= 0,032054
Kết quả từ véc-tơ lỗi được tái tạo	= 0,03204

Cho thấy, ba xác xuất lỗi gần bằng nhau. Lưu ý rằng các xác suất lỗi phù hợp với điểm đã cho trên hình 18.5 đối với $K = 1$ (đường cong ở giữa) và $E_b/N_0 = 5$ dB.

Chương trình thứ 3: NVD18_cdmahmm3

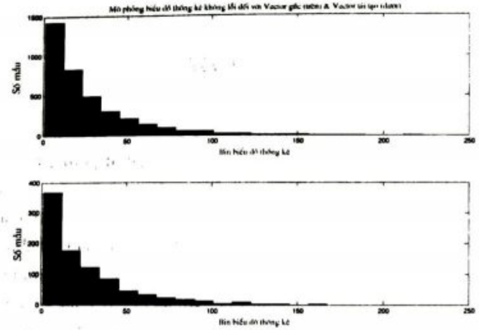
Chương trình cho phép so sánh chuỗi lỗi ban đầu được tạo ra bởi mô phỏng CDMA và chuỗi lỗi được tạo ra bởi mô hình bán Markov. Ta sử dụng hai so sánh ở dạng đồ thị. *Thứ nhất*, so sánh $\Pr\{0^m|1\}$ cả hai chuỗi lỗi như đã thực hiện ở chương 15. *Thứ hai*, so sánh hoành đồ của cả hai chuỗi. Mã chương trình Matlab được cho bởi file **NVD18_cdmahmm3.m**.

Kết quả khi chạy chương trình **NVD18_cdmahmm3.m** được minh họa ở hình 18.6 và hình 18.7. Trong cả hai hình này, các kết quả từ mô phỏng hệ thống CDMA được cho ở khung trên và kết quả từ mô hình bán Markov được cho ở khung bên dưới. Lưu ý về lấy tỷ lệ đối với các chiều cao các thanh trong hình 18.7. Sự khác nhau về lấy tỷ lệ là do mô phỏng CDMA được thực hiện cho 100.000 ký hiệu và kiểm tra mô hình bán Markov sử dụng 25.000 ký hiệu. Vì vậy, khác nhau về tỷ lệ xích là 4.



Hình 18.6: $\Pr\{0^m|1\}$ *đối với véc-tơ lỗi gốc (khung trên) và* $\Pr\{0^m|1\}$ *đối với véc-tơ lỗi được tái tạo (khung dưới)*

Ba lý do chính lý giải cho sự khác nhau trong các khung trên và khung dưới trong hình 18.6 và 18.7. *Thứ nhất*, HMM phải trình bày chính xác kênh ban đầu. Dẫn đến chuỗi số liệu được rút ra từ mô hình phải đủ dài và số vòng lặp đủ lớn để đạt được hội tụ. *Thứ hai*, phải dùng nhiều trạng thái thích hợp. *Thứ ba*, một khi mô hình được rút ra, nó phải được kiểm tra bằng cách xử lý số lượng các ký hiệu đủ lớn nhằm đạt được các kết quả thống kê tin cậy.



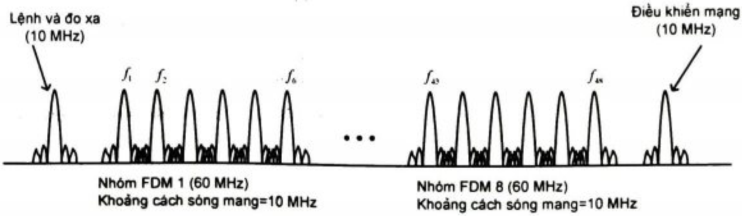
Hình 18.7: Biểu đồ thống kê véc-tơ lỗi gốc (khung trên) và đối với véc tơ lỗi được tái tạo (khung dưới)

18.3. Hệ thống FDM với bộ phát đáp vệ tinh phi tuyến

Mục đích cơ bản của mô phỏng này là nghiên cứu mô phỏng, minh hoạ một vài mô hình quan trọng và kỹ thuật mô phỏng. Các khái niệm quan trọng được minh hoạ trong hệ thống bao gồm việc biểu diễn thông thấp cho các tín hiệu FDM, các mô hình phi tuyến AM/PM & AM/AM cho các bộ khuếch đại công suất lớn và ước tính BER bán giải tích.

18.3.1. Mục đích mô phỏng và mô tả hệ thống

Hệ thống sẽ được mô phỏng là một tuyến truyền thông trong mạng vệ tinh bao gồm 48 trạm mặt đất gửi số liệu tốc độ cao tới 8 trung tâm số liệu vùng. Mỗi trung tâm số liệu xử lý 6 kênh số liệu được gọi là một nhóm FDM. Loại điều chế được dùng cho mỗi kênh là QPSK tại tốc độ ký hiệu 8 M ký hiệu/giây. Tín hiệu được lọc bởi bộ lọc SQRC với hệ số dốc (*roll-off*) là 20%, được khuếch đại bởi bộ khuếch đại công suất tuyến tính, và được phát lên vệ tinh (đường lên). Việc truy cập vệ tinh của 48 trạm mặt đất sử dụng phương pháp FDMA trong đó mỗi trạm mặt đất được ấn định một tần số sóng mang khác bởi bộ điều khiển mạng. Các sóng mang được phân tách bởi băng tần bảo vệ 400 kHz và khoảng cách sóng mang là 10Mz. Hoạch định tần số cho bộ phát đáp 48 kênh được minh hoạ trong hình 18.8.



Hình 18.8: Hoạch định tần số cho hệ thống phát đáp 48 kênh

Toàn bộ 48 tín hiệu đường lên được thu bằng một ăng ten đường lên ở vệ tinh. Tín hiệu thu tại vệ tinh được biểu diễn là:

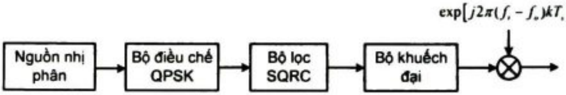
$$x(t) = \sum_{i=1}^{48} a_i(t-\tau_i)\cos\left[2\pi f_i(t-\tau_i) + \phi_i(t-\tau_i)\right] \quad (18.10)$$

Có thể được viết là:

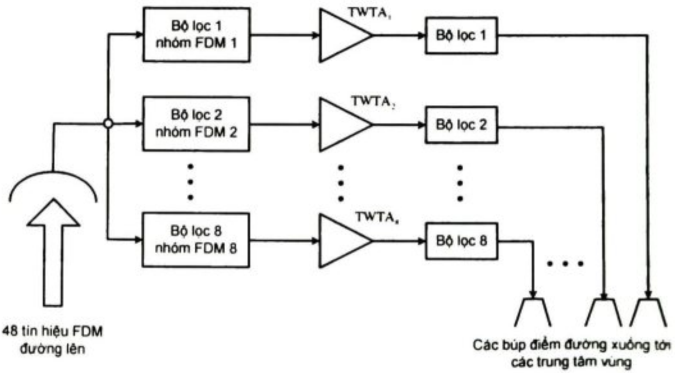
$$x(t) = \mathrm{Re}\left\{\sum_{i=1}^{48} a_i(t-\tau_i) e^{j\phi(t-\tau_i)} e^{-j2\pi f_i \tau_i} e^{j2\pi(f_i - f_0)t} e^{j2\pi f_0 t}\right\} \quad (18.11)$$

Trong đó f_o là tần số tham chiếu được dùng để xác định toàn bộ đường bao phức. Đường bao phức thông thấp từ trạm mặt đất thứ i là tín hiệu QPSK băng tần cơ sở phức $a_i(t).e^{j\phi_i(t)}$. Nó được phát tại tần số sóng mang f_i. Máy phát đường lên cho trạm mặt đất thứ i được minh hoạ trong hình 18.9. Thời gian trễ τ_i thể hiện trễ truyền sóng của kênh đường lên thứ i. Từ (18.11) đường bao phức thông thấp của x(t) là:

$$\tilde{x}(t) = \sum_{i=1}^{48} a_i(t-\tau_i) e^{j\phi(t-\tau_i)} e^{j2\pi f_i \tau_i} e^{j2\pi(f_i - f_0)t} \quad (18.12)$$



Hình 18.9: Máy phát thứ i (đường lên)



Hình 18.10: Chi tiết hóa bộ phát đáp vệ tinh

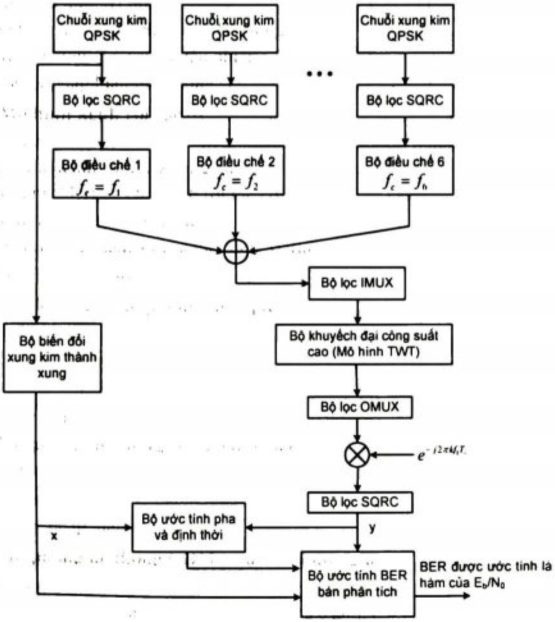
Các tín hiệu đường lên được lọc bằng các bộ lọc đầu vào trên vệ tinh, như được thấy ở hình 18.10, tạo thành 8 nhóm FDM như được thấy trong hình 18.8. Mỗi nhóm được khuếch đại bằng bộ khuếch đèn sóng chạy công suất lớn TWTA. Sau đó các tín hiệu được lọc một lần nữa để khử phổ ngoài băng và méo điều chế ký sinh IM. Mỗi nhóm sóng mang sau đó được truyền

từ vệ tinh tới các trung tâm số liệu vùng qua 8 búp sóng đường xuống với mỗi búp sóng được gửi tới một trung tâm vùng.

Để cung cấp công suất tối đa trong đường xuống, các bộ khuếch đại TWT hoạt động gần với điểm bão hòa. Tuy nhiên, vì mỗi TWTA khuếch đại 6 sóng mang FDM nên méo điều chế ký sinh IM sẽ ảnh hưởng đáng kể lên hiệu năng BER đường xuống. Nhiễu kênh lân cận ACI cũng là nhân tố ảnh hưởng hiệu năng tuyến. Mục đích của nghiên cứu này là mô phỏng một trong số đường xuống (gồm một nhóm 6 sóng mang FDM) từ vệ tinh tới trung tâm số liệu vùng và đánh giá ảnh hưởng của tạp âm nhiệt, méo điều chế ký sinh IM, ISI do các bộ lọc và ACI lên hiệu năng đường xuống.

18.3.2. Mô hình mô phỏng tổng thể

Mô hình mô phỏng được minh hoạ trong hình 18.11. Tạo tín hiệu FDM bằng cách sắp xếp 6 luồng bit nhị phân ngẫu nhiên lên các ký hiệu QPSK, sau đó điều chế chúng tại các tần số sóng mang tương ứng. Các bước khởi tạo trong việc tạo mô hình mô phỏng trường hợp này như sau:



Hình 18.11: Mô hình mô phỏng hệ thống truyền thông vệ tinh

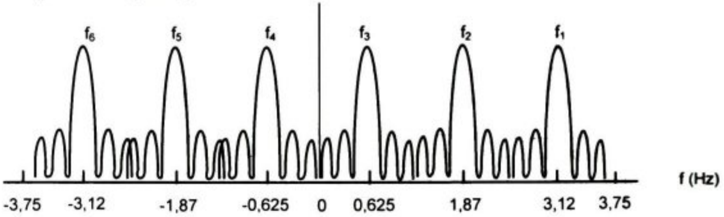
1. Lấy tỉ lệ tần số và thời gian: Tốc độ ký hiệu được chia tỷ lệ xích 8 (10^6), nghĩa là tốc độ ký hiệu được chuẩn hoá thành 1 ký hiệu/giây sao cho tất cả các tốc độ và các tần số được

định rõ trong đơn vị Hz chứ không phải là MHz và các đơn vị thời gian là giây (s) chứ không phải là μs.

2. Tần số tham chiếu để biểu diễn tương đương thông thấp: Biểu diễn tương đương thông thấp của tín hiệu và phần tử hệ thống được dùng trong mô phỏng. Sáu sóng mang FDM được phân cách bởi một khoảng $\Delta f = 1{,}25$ Hz. Với mục đích mô phỏng ta sẽ sử dụng một tần số tham chiếu bằng 0 tương ứng với tần số trung tâm của nhóm 6 sóng mang sao cho các tần số sóng mang (dịch) trong biểu diễn tương đương thông thấp sẽ là:

$$\begin{aligned} f_1 &= \frac{\Delta f}{2}, & f_4 &= -f_1 \\ f_2 &= \Delta f + \frac{\Delta f}{2}, & f_5 &= -f_2 \\ f_3 &= 2\Delta f + \frac{\Delta f}{2}, & f_6 &= -f_3 \end{aligned} \tag{18.13}$$

Được minh hoạ trong hình 18.12



Hình 18.12: Biểu diễn tương đương thông thấp 6 sóng mang FDM

3. Tốc độ lấy mẫu: Tốc độ lấy mẫu được chọn dựa trên toàn bộ độ rộng băng tần của biểu diễn tương đương thông thấp (được cho ở hình18.12) là 3,75 Hz. Theo đó, tốc độ lấy mẫu là $16 \times 3{,}75 = 60$ là thoả đáng. Vì tốc độ ký hiệu là 1 ký hiệu/giây đối với mỗi nguồn tín hiệu QPSK nên tốc độ lấy mẫu 64 mẫu trên ký hiệu được chọn cho tất cả các mô phỏng.

4. Phương pháp ước tính BER: Do tạp âm trong đường xuống là tạp âm cộng, Gausơ và máy thu tuyến tính, nên khi này ta có thể sử dụng bộ ước tính bán giải tích. Vì vậy ta sử dụng bộ ước tính BER QPSK bán giải tích đã đề cập trong chương 10.

5. Thời gian mô phỏng: Với ước tính tỉ số lỗi bán giải tích, thời gian mô phỏng được xác định bởi số lượng các ký hiệu cần thiết để mô phỏng ISI và nhiễu kênh lân cận ACI. Vì sử dụng các bộ lọc SQRC, nên ISI chủ yếu do các bộ lọc ghép kênh đầu vào IMUX và đầu ra OMUX băng khá rộng với đáp ứng tần số gần lý tưởng. Do đó, không cần thiết phải mô phỏng chuỗi đầu vào quá dài. Ta chọn độ dài mô phỏng là 512 ký hiệu trong đó 256 ký hiệu đầu dùng để hiệu chuẩn và 256 ký hiệu còn lại dùng để ước tính BER bán giải tích.

Một khi toàn bộ những giá trị tham số mô phỏng được thiết lập, ta tiến hành lập mô hình mô phỏng cho máy phát, các bộ lọc, bộ khuếch đại phi tuyến, bộ giải điều chế và bộ ước tính bán giải tích.

18.3.3. Tạo tín hiệu FDM đường lên

Nhóm 6 sóng mang FDM đường lên từ 6 trạm mặt đất có thể được mô hình hóa như tín hiệu đến từ một trạm mặt đất có các pha, trễ bất kỳ tương ứng với mỗi sóng mang. Các tín hiệu thể hiện cho lưu lượng được tạo ra bằng cách xắp xếp các ký hiệu số liệu từ mỗi trạm mặt đất vào một chòm sao QPSK, qua bộ lọc SQRC và điều chế tần số sóng mang tương ứng. Thực thi bộ lọc SQRC trong miền thời gian bởi bộ lọc FIR bằng cách dùng phiên bản đáp ứng xung kim được cắt xén và được lấy mẫu, được cho bởi:

$$h_R(t) = (8\beta)\frac{\cos\left[(R+2\beta)\pi t\right] + (8\beta t)^{-1}\sin\left[(R+2\beta)\pi t\right]}{\pi\sqrt{R}.\left[1-(8\beta t)^2\right]} \tag{18.14}$$

Trong đó $R = 1$ và $\beta = 0{,}2$. Đáp ứng xung kim được cắt gọn trong khoảng thời gian 8 ký hiệu. Vì thực thi bộ lọc bằng cách dùng *đáp ứng xung kim*, nên ta đưa vào bộ lọc là *chuỗi xung kim* chứ không phải là *chuỗi xung*.

Sáu tín hiệu sau khi được lọc tạo thành một nhóm FDM được điều chế, và được cộng để tạo tín hiệu FDM đường lên (xem hình 18.11). Để thể hiện trễ truyền sóng ngẫu nhiên từ trạm mặt đất tới vệ tinh, ta đưa lệch pha và trễ thời gian cho mỗi sóng mang. Các hoạt động này được tóm tắt như sau:

1. Tín hiệu QPSK thứ k được định nghĩa là:

$$s_k[n] = \sum_{m=1}^{N}\left(A_{km} + jB_{km}\right)\times\delta\left(\frac{n}{f_s} - kT_s - \tau_k\right) \tag{18.15}$$

Trong đó A_{km} và B_{km} là thành phần đồng pha và vuông pha của đầu vào thứ k, $1 \le k \le 6$, T_s là thời gian ký hiệu, τ_k là thời gian trễ, N là tổng số các ký hiệu trong chuỗi, f_s là tần số lấy mẫu.

2. Tín hiệu lọc thứ k được định nghĩa là:

$$x_k[n] = s_k[n] \otimes p[n] \tag{18.16}$$

Trong đó *p[n]* là đáp ứng xung kim của bộ lọc SQRC, được xác định bởi (18.14), và $\otimes$ là phép tích chập.

3. Tín hiệu điều chế thứ k được định nghĩa là:

$$y_k[n] = a_k \otimes x_k[n] \otimes e^{j\left(\frac{2\pi n f_k}{f_s} + \theta_k\right)} \tag{18.17}$$

Trong đó a_k và f_k là biên độ và tần số của sóng mang thứ k; θ_k thể hiện lệch pha ngẫu nhiên

4. Xác định tín hiệu FDM là:

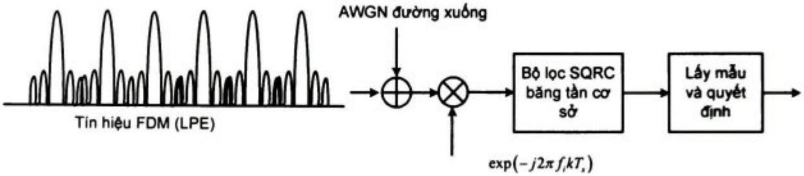
$$z[n] = \sum_{k=1}^{6} y_k[n] \tag{18.18}$$

18.3.4 Mô hình bộ phát đáp vệ tinh

Bộ phát đáp vệ tinh bao gồm: một bộ lọc ghép kênh đầu vào IMUX để cách ly từng nhóm trong 8 nhóm sóng mang, một bộ khuếch đại TWT cho mỗi nhóm 6 sóng mang FDM, và một bộ lọc ghép kênh đầu ra OMUX. Cần phải dịch phổ tần tại vệ tinh vì các tần số đường xuống và đường lên thường là khác nhau, ta coi việc dịch phổ tần là lý tưởng. Thực thi các bộ lọc IMUX và OMUX trong mô phỏng bằng cách dùng các bộ lọc Butterworth bậc bốn có độ rộng băng tần 3 dB là 4 Hz. ISI do các bộ lọc này là tối thiểu. Dùng các phần tử phi tuyến không nhớ PM-AM và AM- AM có độ lợi đơn vị để mô hình hóa bộ khuếch đại TWT như đã được đề cập trong chương 12. Coi các đặc tính PM-AM và AM-AM được cho ở dạng bảng, "*độ lùi*" là một tham số của mô hình.

18.3.5. Mô hình máy thu và bộ ước tính BER bán giải tích

Sáu sóng mang trong mỗi nhóm FDM được giải điều chế riêng biệt tại máy thu trạm mặt đất tại mỗi trung tâm vùng. Xét ước tính BER cho một trong số các sóng mang FDM và giải điều chế của bất kỳ sóng mang nào trong 6 sóng mang bằng cách dịch tần số, theo sau là lọc SQRC như được thấy ở hình 18.13. Cần phải chuẩn pha và chuẩn định thời để giải điều chế và tách tín hiệu. Chúng được thực thi bởi các phân hệ đồng bộ trong máy thu, ở đây ta không mô phỏng tường minh các khối chức năng này mà giả thiết là đã đồng bộ hoàn hảo.



Hình 8.13: Mô hình máy thu trạm mặt đất

Vì máy thu tuyến tính và tạp âm đường xuống là tạp âm Gausơ cộng tại đầu vào máy thu, nên ta dùng bộ ước tính tỉ số lỗi bán giải tích để ước tính ảnh hưởng của ISI do lọc và các ảnh hưởng của bộ khuếch đại phi tuyến gồm méo điều chế ký sinh IM và tạp âm đường xuống. Cần phải hoàn tất các hiệu chuẩn trước khi ước tính tỉ số lỗi là một hàm của E_b / N_o đường xuống. Các hoạt động này bao gồm:

1. Ước tính E_b tại đầu vào máy thu cho mỗi sóng mang.

2. Lấy tương quan chéo dạng sóng vào/ra, ước tính lệch pha và định thời từ hàm tương quan chéo.

3. Ước tính độ rộng băng tần tạp âm của máy thu bằng cách đưa một xung kim vào máy thu, lấy bình phương và cộng các giá trị của đáp ứng xung kim để đạt được độ rộng băng tần tạp âm tương đương được thiết kế theo (10.12).

4. Làm trễ dạng sóng đầu vào một cách thích hợp trước khi ước tính tỉ số lỗi.

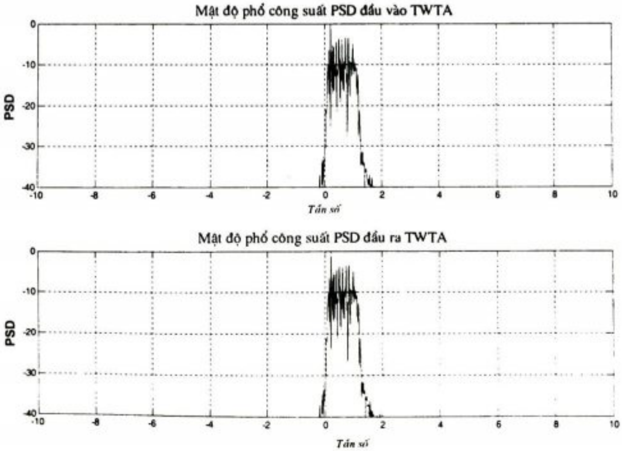
18.3.6. Các kết quả mô phỏng

Trong phần này ta trình bày các kết quả mô phỏng nhằm minh hoạ các ảnh hưởng méo phi tuyến và tạp âm Gausơ cộng lên hiệu năng BER của hệ thống FDM. Chương trình mô phỏng được cho bởi file **NVD18_nlsat.m** trong phụ lục 18A. Mục đích của mô phỏng là minh hoạ các ảnh hưởng của mô hình TWT phi tuyến, ACI và tạp âm cộng. Các kết quả mô phỏng ở dạng đồ thị bao gồm: đồ thị mật độ phổ công suất PSD tại đầu vào/đầu ra của mô hình TWT, chòm sao tín hiệu minh hoạ các ảnh hưởng phi tuyến và BER do tạp âm và méo phi tuyến. (Vì dùng ước tính BER bán giải tích nên tán xạ điểm được quan sát trong chòm sao tín hiệu là do tính phi tuyến, ACI và lọc gây ra. Các ảnh hưởng tạp âm không được bao gồm trong chòm sao tín hiệu). Ở đây, ta nghiên cứu chi tiết hai kết quả mô phỏng. Bằng cách thay đổi biên độ sóng mang, khoảng cách sóng mang, độ lùi đầu vào TWT, cho phép dễ dàng khảo sát một số ảnh hưởng khác.

a) Xác nhận tính hợp lệ ranh giới

Cần phải đặc biệt lưu ý "*kiểm tra tính đúng đắn*" của mô phỏng đường ranh giới sao cho các khối chức năng trong mô hình mô phỏng và phương pháp luận tổng thể được hợp lệ. Trong tất cả các mô phỏng dưới đây, mô hình và phương pháp luận vẫn như trước, chỉ thay đổi giá trị tham số. Theo đó, mô phỏng tính hợp lệ đường ranh giới sẽ thiết lập độ tin cậy cho tất cả các mô phỏng tiếp theo. Điều này được thực hiện một cách dễ dàng bằng cách mô phỏng cho một kênh đơn và đảm bảo rằng mô hình TWT hoạt động trong vùng tuyến tính. Bằng cách đặt:

$$a_k = \begin{cases} 1, & k = 1 \\ 0, & k = 2,3,4,5,6 \end{cases} \tag{18.19}$$

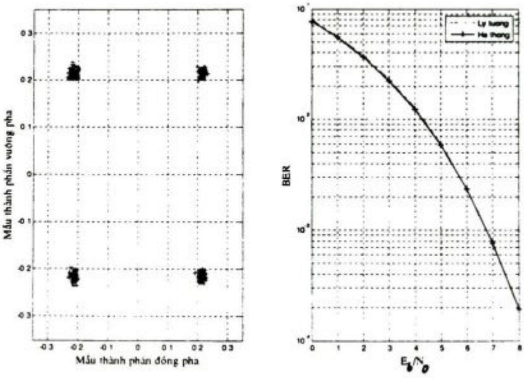


Hình 18.14: PSD tại đầu vào và đầu ra của mô hình TWT (độ lùi đầu vào là -20dB)

Ta đảm bảo không xẩy ra ACI. Bằng cách cho hoạt động với độ lùi đầu vào là -20 dB cho phép bỏ qua méo phi tuyến trong độ rộng băng tần của tín hiệu kênh 1. Theo đó, thể hiện rõ chỉ có ảnh hưởng tạp âm Gausơ cộng làm suy thoái hiệu năng và kết quả. (Lưu ý rằng bỏ qua ISI do các bộ lọc IMUX và OMUX).

PSD tại đầu vào và đầu ra của mô hình TWT làm việc tại độ lùi —20 dB, được minh hoạ trong hình 18.14. Thực ra chỉ có một kênh hoạt động rõ ràng. Ngoài ra, bỏ qua nở phổ do điều chế ký sinh, nó minh hoạ cho TWT đang làm việc trong vùng gần như tuyến tính.

Chòm sao tín hiệu không bị ảnh hưởng bởi tạp âm được minh hoạ ở nửa bên trái hình 18.15. Thấy rõ, chòm sao tín hiệu ít bị tán xạ minh chứng cho ảnh hưởng phi tuyến là rất nhỏ, thể hiện sự tồn tại ISI là rất nhỏ. Các điểm tín hiệu trong chòm sao tín hiệu được sử dụng để tạo hiệu năng BER được minh hoạ ở nửa bên phải hình 18.15. *Lưu ý rằng, sự khác nhau giữa các kết quả mô phỏng và các kết quả lý tưởng (lý thuyết) là không đáng kể. Vì vậy, mô phỏng đường ranh giới được hợp lệ.*



Hình 18.15: Chòm sao tín hiệu do ảnh hưởng phi tuyến và BER do tạp âm

b) Ảnh hưởng tạp âm và phi tuyến - 5 sóng mang FDM

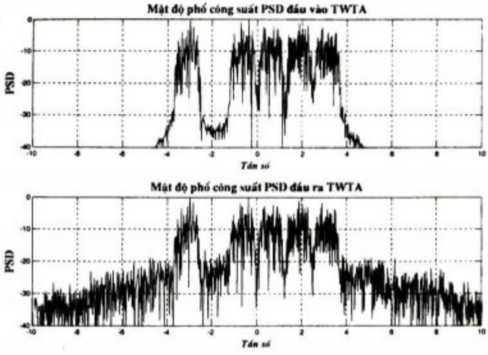
Trong mô phỏng này, biên độ của sóng mang thứ năm được đặt bằng 0, và biên độ của tất cả các sóng mang khác được đặt bằng 1. (Sóng mang thứ 5 được chọn để tạo phổ có một khoảng trống, ngoài ra, để đảm bảo rằng các kênh này gây ra ACI đáng kể lên kênh 1 hoạt động). Độ lùi đầu vào của TWT được đặt là -5 dB, để minh hoạ hoạt động phi tuyến tức thời của mô hình TWT. Nói cách khác, ta dùng các tham số:

$$a_k = \begin{cases} 0, & k = 5 \\ 1, & k = 1,2,3,4,6 \end{cases} \tag{18.20}$$

$$\text{ibo} = -5 \tag{18.21}$$

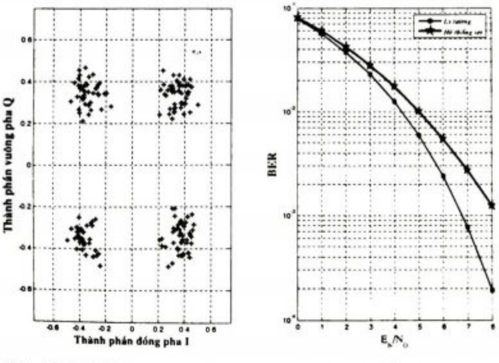
Phổ tại đầu vào/ra của TWT được minh hoạ ở hình 18.16. Lưu ý rằng, khoảng trống trong PSD đầu vào tại vị trí phổ được xác định bởi f_5. PSD tại đầu ra TWT cho thấy do tính phi tuyến

gây méo điều chế ký sinh IM dẫn đến làm lấp đầy khoảng trống phổ. Hiện tượng "*làm đầy*" này được gọi là gia tăng phổ và là nguồn gây lỗi chủ yếu trong hệ thống FDM làm việc có tính phi tuyến. Để được rõ hơn, ta nên chạy lại mô phỏng với độ lùi đầu vào TWT là -20 dB, khi này TWT hoạt động như một bộ khuếch đại gần như tuyến tính và chỉ ra rằng gia tăng phổ được giảm nhẹ đáng kể.



Hình 18.16: PSD tại đầu vào và đầu ra của mô hình TWT (độ lùi đầu vào là -5dB)

Các ảnh hưởng của ISI, ACI, tạp âm và méo do TWT được minh hoạ trong hình 18.17. Chòm sao tín hiệu, minh hoạ tán xạ điểm chủ yếu do các ảnh hưởng của phi tuyến (do dùng độ lùi trong mô hình TWT là nhỏ), được minh hoạ ở nửa bên trái của hình 18.17. Bộ ước tính BER bán giải tích dùng các điểm tín hiệu này để xác định BER toàn bộ hệ thống, được minh hoạ ở nửa bên trái của hình 18.17. Thấy rõ ảnh hưởng của thiết bị phi tuyến làm suy thoái hiệu năng BER.



Hình 18.17: Chòm sao tín hiệu và BER (độ lùi đầu vào là -5dB và chòm sao tín hiệu $E_b/N_0 = 8dB$)

THUẬT NGỮ VIẾT TẮT

3G	Third Generation Mobile Communications System	Hệ thống thông tin di động thế hệ thứ ba
4G	Fourth Generation Mobile Communication System	Hệ thống thông tin di động thế hệ thứ tư
ACI	Adjacent Cell Interference	Nhiễu ô lân cận
AM-AM	Amplitude-to-Amplitude Transfer Characteristic	Đặc tính truyền đạt biên độ-biên độ
AM-PM	Amplitude-to-Phase Transfer Characteristic	Đặc tính truyền đạt biên độ-pha
AMPS	Advanced Mobile Phone System	Hệ thống điện thoại di động tiên tiến
AOA	Angle Of Arrival	Góc tới
AOD	Angle Of Departure	Góc xuất phát
AQAM	Adaptive QAM	Điều chế QAM thích ứng
AWGN	Additive White Gaussian Noise	Tạp âm Gausơ trắng cộng
BER	Bit Error Rate	Tỉ Lệ lỗi bit
BLER	Block Error Rate	Tỷ số lỗi khối
BP	Bandpass	Thông giải hay dải thông
bps	bit per second	bit trên giây
BPSK	Binary Phase Shift Keying Modulation	Điều chế khóa dịch pha nhị phân
BS	Base Station	Trạm gốc
BSC	Binary Symmetric Channel	Kênh nhị phân đối xứng
CAD	Computer—Aided Design	Thiết kế được hỗ trợ bởi máy tính
CC	Convolutional Code	Mã xoắn
CCI	Co channel Interference	Nhiễu đồng kênh
CDF	Cumulative Distribution Function	Hàm phân bố tích lũy
CDMA	Code Division Multiple Access	Đa truy nhập phân chia theo mã
CIR	Carrier Interference Ratio	Tỷ số sóng mang trên nhiễu
CIS	Conventional Importance Sampling	Lấy mẫu quan trọng truyền thống

CLT	Central Limit Theory	Lý thuyết giới hạn trung tâm
CNR	Carrier to Noise Ratio	Tỷ số sóng mang trên tạp âm
CRC	Cyclic Redundancy Code	Mã dư vòng
CSI	Channel State Information	Thông tin trạng thái kênh
CSMP	Continuous System Modeling Program	Chương trình mô hình hóa hệ thống liên tục
DCM	Discrete Channel Model	Mô hình kênh rời rạc
DF	Direct Form	Dạng trực tiếp
DF II Filter	Direct Form II Filter	Bộ lọc dạng trực tiếp II
DFE	Decision Feedback Equalizer	Bộ cân bằng hồi tiếp quyết định
DFT	Discrete Fourier Transform	Biến đổi Fourier rời rạc
DL	Downlink	Đường xuống
DMS	Discrete Memoryless Source	Nguồn rời rạc không nhớ
DMS	Discrete Memoryless Source	Nguồn không nhớ rời rạc
DPS	Delay Power Spectrum	Phổ công suất trễ
DPSK	Differential Phase Shift Keying	Khoá dịch pha vi sai
DQPSK	Differential Quadrature Phase Shift Keying	Điều chế khóa dịch pha cầu phương vi sai
DS	Direct Sequence	Chuỗi trực tiếp
DSP	Digital Signal Processing	Xử lý tín hiệu số
DSSS	Direct Sequence Spread Spectrum	Trải phổ chuỗi trực tiếp
Eb	Bit Engery	Năng lượng Bit thông tin
FDD	Frequency Division Duplex	Song công phân chia theo tần số
FDM	Frequency Division Multiplex	Ghép kênh phân chia theo tần số
FDMA	Frequency Division Multiple Access	Đa truy nhập phân chia theo tần số
FER	Frame Error Rate	Tỉ số lỗi khung
FFT	Fast Fourier Transform	Biến đổi Fourier nhanh
FIR	Finite-Duration Impulse Response	Đáp ứng xung hữu hạn
FM	Frequency Modulation	Điều tần
FSK	Frequency Shift Keying	Điều chế khoá dịch tần
GSM	Global System for Mobile Telecommunications	Hệ thống thông tin di động toàn cầu

HMM	Hidden Markov Model	Mô hình Markov ẩn
HPA	High-Power Amplifier	Bộ khuếch đại công suất cao
IBO	Input BackOff	Độ lùi đầu vào
ICI	Inter Carrier Interference	Nhiễu giữa các sóng mang
IDD	Idependent dent and Identically Distributed	Phân bố đồng nhất và độc lập
IFFT	Inverse Fast Fourier Transform	Biến đổi Fourier nhanh đảo
IIR	Infinite-Duration Impulse Response	Đáp ứng xung vô hạn
IIS	Improved Importance Sampling	Lấy mẫu quan trọng cải tiến
IM	Intermodulation	Điều chế giao thoa
IMUX	Input Multiplex	Ghép kênh đầu vào
I-Q Plane	Inphase-Quadrature Plane	Mặt phẳng đồng pha-vuông pha
ISI	Inter-Symbol Interference	Giao thoa giữa các ký hiệu
LCG	Linear Congruence Generator	Bộ tạo số đồng dư tuyến tính
LOS	Line of Sight	Đường truyền thẳng
LP	Lowpass	Thông thấp
LTIV	Linear Time-Invariant	Hệ thống tuyến tính bất biến
LTV	Linear Time-Varying	Tuyến tính thay đổi theo thời gian
MA	Multiple Access	Đa truy nhập
MAI	Multiple Access Interference	Nhiễu đa người sử dụng
MC	Monte Carlo	Phương pháp Monte Carlo
MC-CDMA	Multi Carrier Code Division Multiple Access	Đa truy nhập phân chia theo mã đa sóng mang
MIMO	Multiple Input Multiple Output	Nhiều đầu và nhiều đầu ra
ML	Maximum Likelihood	Khả giống cực đại
MMSE	Minimum Mean Square Error	Sai lỗi bình phương trung bình cực tiểu
M-PSK	M-array Phase-Shift Keying	Điều chế khóa dịch pha M trạng thái
MQAM	Multilevel-QAM	QAM nhiều mức
MS	Mobile Station	Trạm di động (máy di động)
MSC	Mobile Services Switching Center	Trung tâm chuyển mạch các dịch vụ di động Tổng đài di động
MSE	Mean-Square Error	Lỗi bình phương trung bình

MSK	Minimum Shift Keying	Khóa dịch pha cực tiểu
MUI	Multi-User Interference	Nhiễu đa truy nhập
NLDE	Nonlinear Differential Equation	Phương trình vi phân phi tuyến
NRP	Normalized Received Power	Công suất thu chuẩn hóa
NRZ	Non-Return-to Zero	Không trở về không
OBO	Output BackOff	Độ lùi đầu ra
OFDM	Orthogonal Frequency Division Multipex	Ghép kênh phân chia theo tần số trực giao
OFDMA	Orthogonal Frequency Division Multiple Access	Đa truy nhập phân chia theo tần số trực giao
OMUX	Output Multipex	Ghép kênh đầu ra
OQPSK	Offset Quadrature Phase Shift Keying	Điều chế khoá dịch pha cầu phương lệch
pdf	probability density function	Hàm mật độ xác suất
PDP	Power Delay Profile	Lý lịch trễ công suất
PG	Processing Gain	Độ lợi xử lý
PLL	Phase-Locked Loop	Vòng khóa pha
PM	Phase Modulation	Điều pha
PN	Pseudo Noise	Giả tạp âm
PN Chip	Pseudo Noise Chip	Chip PN
PN Sequence	Pseudo Noise Sequence	Chuỗi giả tạp âm
PSD	Power Spectral Density	Mật độ phổ công suất
PSK	Phase-Shift-Keying	Khóa dịch pha
PSTN	Public Switched Telephone Network	Mạng điện thoại chuyển mạch công cộng
QAM	Quadrature Amplitude Modulation	Điều chế biên độ cầu phương
QoS	Quality of Service	Chất lượng dịch vụ
QPSK	Quadrature Phase Shift Keying Modulation	Khóa dịch pha cầu phương
RDS	Root mean square Delay Spread	Trải trễ trung bình quân phương
RF	Radio Frequency	Tần số vô tuyến
RF Carrier		Sóng mang vô tuyến
rms	Root mean square	Trung bình quân phương

RX	Receiver	Máy thu
SA	Semianalytic	Phương pháp bán giải tích
SDMA	Space Division Multiple Access	Đa truy nhận phân chia theo không gian
SER	Symbol Error Rate	Tỷ lệ lỗi ký hiệu
SF	Spreading Factor	Hệ số trải phổ
SINR	Signal to Interference plus Noise Ratio	Tỷ số tín hiệu trên nhiễu cộng tạp âm
SIR	Signal to Interference Ratio	Tỷ số tín hiệu trên nhiễu
SNR	Signal to Noise Ratio	Tỷ số tín hiêu trên tạp âm
SQRC	Square Root Raised Cosine	Bộ lọc Cosine tăng căn bình phương
SS	Strictly Stationary	Quá trình dừng chặt
SS	Spread Spectrum	Trải phổ
TDD	Time Division Duplex	Song công phân chia theo thời gian
TDL	Tap Delay Line	Đường trễ rẽ nhánh
TDMA	Time Division Multiple Access	Đa thâm nhập phân chia theo thời gian
Transposed DF II Filter	Transposed Direct Form II Filter	Bộ lọc dạng trực tiếp II chuyển vị
TWTA	Traveling Wave Tube Amplifier	Bộ khuếch đại đèn sóng chạy
TX	Transmitter	Máy phát
US	Uncorrelated Scatering	Tán xạ không tương quan
VCO	Voltage Controlled Oscillator	Bộ dao động được điều khiển bằng điện áp
WSS	Wide Sense Stationary	Quá trình dừng nghĩa rộng
WSSUS	Wide Sense Stationary and Uncorrelated Scatering	Tán xạ không tương quan dừng nghĩa rộng
ZF	Zero Forcing	Cưỡng bức về không

TÀI LIỆU THAM KHẢO

1. Michel C. Jeruchim, Philip Balaban, K. Sam Shanmugan, K. Sam Shanmugan: Simulation of Communication Systems: *Modeling, Methodology, and Techniques, Second Edition*, New York, 2002.
2. Matthias Patzold: *Mobile Fadinh Channel, John Wiley&Sons, 2002.*
3. John G.Proakis: *Contemporary Communication System using Matlab*, Bill Stenquist, 2000.
4. John G.Proakis: *Digital Signal Processing Using Matlab V4*, Bill Stenquist, 1997.
5. Steven M. Kay: *Fundamentals Statistical Signal Processing Estimation*, Prentice Hall PTR,1998.
6. Alfred Mertins: *Signal Analysis, Wavelets, Filter Banks,Time-Frequency Transforms and Applications*, John Willey & Sons,1998
7. Jamal T. Manassah: *Elementary mathematical and Computational tools for electrical and computer engineers using matlab*, CRC Press LLC, 2001.
8. Roy M. Howard: *Principles of Random Signal Analysis and Low Noise Design, Wiley*, Great Britain, 2002.
9. Derek A. Wells: *Simulation of a Wireless Communication Channel to Determine a Best Topology for a Base Station Array Antenna*, Master, Virginia Polytechnic Institute and State University, 2003.
10. Llya M. Sobol: *Primer for the Monte Carlo Method*, CRC Press, Tokyo, 1994
11. Nguyễn Phạm Anh Dũng: *Lý thuyết trải phổ và đa truy nhập vô tuyến*, Nhà xuất bản Bưu điện - 2004.
12. Rappaport, T.S.: *Wireless Communications, Principles and Practice*, 2nd ed., Prentice Hall PTR, 2002
13. Dr. L. Hanzo and other: *Adaptive Wireless Transceiver*, Wiley, Great Britain, 2002
14. Goldsmith, A.J. Chua, S.G.: *Variable-rate variable-power MQAM for fading channels*, IEEE Tr. on Communications, Vol. 45, No. 10. Oct. 1997
15. Che,H., *Adaptive OFDM and CDMA algorithms for SISO and MIMO channels*, Ph.D thesis, Delft University of technology, the Nethelands., 2005.
16. Dimitris N. Kalofonos & John G.Proakis: *Performace of Adaptive MC-CDMA Detectors in Rayidly Fading Rayleigh Channels*, IEEE, 2003.
17. John G..Proakis: *Performance of Adaptive MC-CDMA Derectors in Rapidly Fading Rayleigh Channels*, Proc, IEEE - 2003.

18. Eric Phillip LAWREY BE, *Adaptive Techniques for Multiuser OFDM*, Ph.D thesis, James Cook University, 2002.

19. Fredrik Tufvesson: *Channel Related Optimization of Wireless Communication System*, Reprocentralen, Ph.D thesis, Lund University, Sweden, 1998.

20. Dr. Proakis, John G. 2001: *Digital Comunications*, McGraw-Hill.

21. Dr. Bernard Scalar. 2004: *Digital Communications*, Prentice-Hall.

22. Dr. William C.Y.Lee: *Mobile Communications Design Fundamental*, John Wiley & Sons, Inc, 1996

MỤC LỤC

Lời nói đầu ... 3
Chương 1: Vai trò của mô phỏng ... 5
1.1. Mở đầu.. ... 5
1.2. Minh họa mức độ phức tạp... 5
1.3. Các mặt đa kỷ luật của mô phỏng. ... 10
1.4. Các mô hình.. ... 13
1.5. Mô phỏng tất định và mô phỏng ngẫu nhiên ... 14
1.6. Vai trò của mô phỏng ... 19
1.7. Gói phần mềm để mô phỏng và khuyến nghị ... 22
1.8. Sử dụng Matlab. ... 24
Chương 2: Phương pháp luận mô phỏng ... 27
2.1. Mở đầu.. ... 27
2.2. Các mặt của phương pháp luận ... 29
2.3. Ước tính hiệu năng ... 41
Chương 3: Lấy mẫu và lượng tử ... 44
3.1. Mở đầu. ... 44
3.2. Lấy mẫu ... 45
3.3. Lượng tử ... 52
3.4. Tái tạo và nội suy ... 57
3.5. Tần số lấy mẫu trong mô phỏng ... 62
3.6. Điều chế xung mã PCM ... 69
Chương 4: Mô hình mô phỏng thông thấp cho tín hiệu và hệ thống thông dải ... 76
4.1. Mở đầu. ... 76
4.2. Đường bao phức thông thấp cho tín hiệu thông dải ... 77
4.3. Hệ thống thông dải tuyến tính ... 97
4.4. Tín hiệu đa sóng mang. ... 104
4.5. Hệ thống phi tuyến và thay đổi theo thời gian. ... 106
4.6. Minh họa tín hiệu và hệ thống tuyến tính bất biến.. ... 110
Chương 5: Mô hình bộ lọc và kỹ thuật mô phỏng ... 113
5.1. Mở đầu.. ... 113
5.2. Bộ lọc IIR và FIR ... 115
5.3. Thực hiện bộ lọc IIR và FIR ... 116
5.4. Bộ lọc IIR - Kỹ thuật tổng hợp và đặc điểm ... 120
5.5. Bộ lọc FIR - Kỹ thuật tổng hợp và đặc điểm ... 130

Chương 6: Vòng khóa pha và phương pháp giải phương trình vi phân 145
6.1. Mở đầu.. 145
6.2. Khái niệm vòng khóa pha cơ bản. 146
6.3. Vòng bậc một và bậc hai 151
6.4. Mô phỏng vòng khóa pha PLL. 155
6.5. Dùng mô phỏng để giải phương trình vi phân 161
Chương 7: Tạo và xử lý tín hiệu ngẫu nhiên 168
7.1. Mở đầu.. 168
7.2. Quá trình dừng và Ergodic 169
7.3. Bộ tạo số ngẫu nhiên phân bố đều 173
7.4. Ánh xạ biến ngẫu nhiên phân bố đều thành biến ngẫu nhiên có pdf tùy ý 181
7.5. Tạo số ngẫu nhiên Gausơ không tương quan 188
7.6. Tạo số ngẫu nhiên Gausơ tương quan 193
7.7. Thiết lập hàm mật độ xác suất và mật độ phổ công suất.. 197
7.8. Các bộ tạo chuỗi PN. 198
7.9. Xử lý tín hiệu 201
Chương 8: Hậu xử lý 204
8.1. Mở đầu. 204
8.2. Kỹ thuật đồ họa cơ bản 205
8.3. Ước tính 210
8.4. Mã hóa 220
8.5. Một số ứng dụng điển hình 225
Chương 9: Giới thiệu phương pháp Monte Carlo 235
9.1. Mở đầu.. 235
9.2 Khái niệm cơ bản 235
9.3. Ứng dụng vào hệ thống truyền thông - Kênh AWGN 240
9.4. Tích phân Monte Carlo 247
Chương 10: Mô phỏng Monte Carlo hệ thống truyền thông *253*
10.1. Mở đầu 253
10.2. Hai mô phỏng Monte Carlo điển hình. 254
10.3. Kỹ thuật bán phân tích 260
Chương 11: Phương pháp luận mô phỏng hệ thống vô tuyến 272
11.1. Mở đầu 272
11.2. Tốc độ lấy mẫu và đơn giản hóa mức hệ thống 273
11.3. Phương pháp luận toàn diện. 274
Chương 12: Mô hình hóa và mô phỏng thiết bị phi tuyến 289
12.1. Mở đầu 289
12.2. Mô hình hóa và mô phỏng thiết bị phi tuyến không nhớ 292

12.3. Mô hình hóa và mô phỏng thiết bị phi tuyến có nhớ.......304
12.4. Kỹ thuật giải phương trình vi phân phi tuyến NLDE.......308
12.5. Mô phỏng PLL minh họa thiết bị phi tuyến.......316
Chương 13: Mô hình hóa và mô phỏng hệ thống thay đổi theo thời gian.......318
13.1. Mở đầu.......318
12.2. Các mô hình cho hệ thống tuyến tính thay đổi theo thời gian LTV.......320
13.3. Mô hình quá trình ngẫu nhiên.......329
13.4. Mô hình mô phỏng cho hệ thống LTV.......332
13.5. Thực hiện mô phỏng trên Matlab.......335
Chương 14: Mô hình hóa và mô phỏng kênh dạng sóng.......339
14.1. Mở đầu.......339
14.2. Kênh hữu tuyến và ống dẫn sóng.......342
14.3. Kênh vô tuyến.......343
14.4. kênh pha đinh đa đường.......346
14.5. Mô hình hóa kênh pha đinh đa đường.......352
14.6. Các mô hình quá trình ngẫu nhiên.......354
14.7. Phương pháp luận mô phỏng.......359
Chương 15: Mô hình kênh rời rạc.......375
15.1. Mở đầu.......375
15.2. Mô hình kênh rời rạc không nhớ.......378
15.3. Mô hình Markov cho kênh rời rạc có nhớ.......380
15.4. Mô hình Gilbert và Frichman.......389
15.5. Ước tính các tham số của mô hình Markov.......392
15.6. Hai trường hợp điển hình.......402
Chương 16: Các kỹ thuật mô phỏng hiệu quả.......407
16.1. Mở đầu.......407
16.2. Ngoại suy đuôi.......407
16.3. Bộ ước tính pdf.......409
16.4. Lấy mẫu quan trọng.......411
Chương 17: Mô phỏng hệ thống vô tuyến tế bào.......423
17.1. Mở đầu.......423
17.2. Hệ thống vô tuyến tế bào.......425
17.3. Phương pháp luận mô phỏng.......436
Chương 18: Mô phỏng hệ thống CDMA và FDM đơn giản.......451
18.1. Mở đầu.......451
18.2. Hệ thống đa truy nhập phân chia theo mã CDMA.......451
18.3. Hệ thống FDM với bộ phát đáp vệ tinh phi tuyến.......461
Thuật ngữ viết tắt.......470
Tài liệu tham khảo.......475

Mô phỏng hệ thống viễn thông và ứng dụng **Matlab**

Chịu trách nhiệm xuất bản
LƯU ĐỨC VĂN

Biên tập: TRẦN VŨ THƯỞNG
BÙI NGỌC KHOA - ĐẶNG HUY TOÀN
Chế bản: NGUYỄN MẠNH HOÀNG
Sửa bản in: ĐẶNG HUY TOÀN
Trình bày bìa: TRỊNH THẾ VINH

In 500 bản, khổ 19 x 27 cm, tại Công ty CP Nhà in Khoa học và Công nghệ
Số đăng ký kế hoạch xuất bản 62-2007/CXB/45 - 03/BuĐ
Số quyết định xuất bản: 118/QĐ-NXB BĐ ngày 25/6/2007
In xong và nộp lưu chiểu tháng 7 năm 2007.